All Electron User programs work on BBC Micros with OS 1.2 and Basic II

A Database Publication

electron user

Vol.1 No.12 September 1984 £1

Which sort meets YOUR needs?

Interchange sort

Bubble sort

Shell sort

All the facts about the Electron Plus 1

Keyboard control of graphic characters

Splash out on our logic game

TRAPPED IN A HAUNTED HOUSE

Play it if you dare!

FREE CONTEST
Win the First Byte printer interface

HOW TO TURN YOUR SOFTWARE INTO HARD CASH.

Turning a great idea into a profitable idea isn't easy.

It needs skill, patience and sheer hard work. But to be really profitable, it also needs the backing of a large, respected company.

At British Telecom we are now looking for writers of games and educational programs to help us launch a new and exciting range of software.

If we like your idea we'll send you a cheque straightaway as an advance on royalties.

Then we'll package your program, advertise it and distribute it with the care and attention you'd expect from one of Britain's biggest companies.

So if you'd like to turn your software into hard cash, simply send your program, on cassette or disk, to

FIREFLY
SOFTWARE

British Telecom,
Wellington House, Upper St. Martin's Lane, London WC2H 9DL.
Tel: 01-379 6755

CONTENTS

Vol. 1 No. 12 September 1984

electron user

News
All that's new in the expanding world of the Electron. 6

Beginners
Part 8 of our gentle introduction to Basic deals with loops within loops. 8

Maths Workout
Confused by MOD and DIV? Then let Dave Robinson help you. 10

Notebook
Nested loops in action, controlling sound and graphics. 12

Scrapbook
A collection of simple, short programs for your entertainment and edification. 18

Sounds Exciting
Yet more weird and wonderful Electron sounds. 21

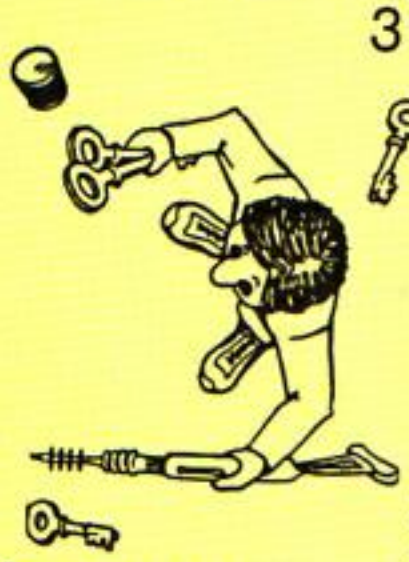
Sailor
Animated nautical manoeuvre as our sailor does the hornpipe. 24

Splash
Think fast and avoid a drenching. 26

Software Surgery
All you want to know about the latest in software from our frank reviewers. 27

Maths Test
Your powers of mental arithmetic put to the test. 31

Which Sort?
Confused by all the different sorts? Paul Hutson sorts it out. 34

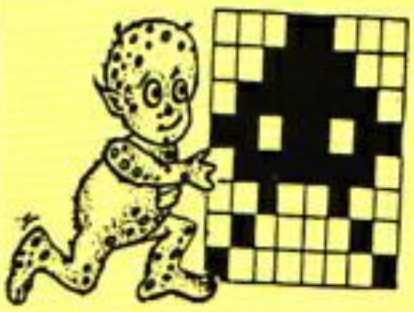


Sorts in action
An animated program showing the sorts as they work. 36

Classroom Invasion
How multicoloured characters invaded a primary school. 38

Haunted House
Avoid the spectres and spooks in our best action game ever. 32

Competition
Two chances to win First Byte's new printer interface in our free competition. 41



Casting Agency
More shapes from our readers to brighten your programs. 45

Program Probe
An in-depth look at on-screen movement. 42

Hardware Review
Acorn's Plus 1 for the Electron comes under scrutiny. 48

Micro Messages
The pages you write yourself. A selection from our mailbag. 61

Managing Editor
Derek Meakin
Features Editor
Pete Bibby
Production Editor
Peter Glover
Layout Design
Heather Sheldrick
Advertisement Manager
John Riding
Advertising Sales
John Snowden
Marketing Manager
Susie Lipman

Published by Database Publications Ltd
Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.
Telephone: 061-456 8383 (Editorial) 061-456 8500 (Advertising)
Subscriptions: 061-480 0171 Telex: 667664 SHARET G. Prestel: 614568383.

Subscription rates for 12 issues, post free:
£12 UK
£13 Eire (IR £16)
£20 Europe
£20 Rest of world (surface)
£40 Rest of world (airmail)

Trade distribution in the UK and overseas: Contact Steve Fletcher, Circulation Manager of Database Publications at the above address, or telephone him on 061-480 4153.
Electron User is an independent publication. Acorn Computers Ltd, manufacturers of the Electron, are not responsible for any of the articles in this issue or for any of the opinions expressed.

Electron User welcomes program listings and articles for publication. Material should be typed or computer-printed, and preferably double-spaced. Program listings should be accompanied by cassette tape or disc. Please enclose a stamped, self-addressed envelope, otherwise the return of material cannot be guaranteed. Contributions accepted for publication will be on an all-rights basis.

FIRST BYTE
ELECTRON JOYSTICK INTERFACE

Printer Interface OUT NOW
Uses normal BBC printer commands
No software required!

ELECTRON JOYSTICK INTERFACE

Electron users! This is the add-on everyone wants. It's the new Electron switched joystick interface from First Byte - available now with *free* conversion tape that vastly extends your game range right away.

The interface operates with all 'Atari-style' 9-pin joysticks, and its many advanced design features put it way out in front for quality and reliability. That's why, to date 15 major software houses are already bringing out games that work directly with the First Byte Electron Joystick Interface - and many more are sure to follow.

FREE conversion tape - play all these top games right now

Every Electron Joystick Interface comes with a free conversion tape, so you can use some of the most popular games around right now:

- Killer Gorilla
- Moonraider
- Positron
- Croaker
- Swoop
- Bandits at 3 o'clock
- Escape from Moonbase Alpha
- Cybertron Mission
- Cylon Attack
- Kamakazi
- Chuckie Egg
- Atom Smasher
- Alien Break In
- Birds of Prey
- Galaxy Wars
- City Defence
- Monsters
- Pool
- Pengwyn
- Lunar Rescue
- Bugblaster
- Blagger
- Bed Bugs
- Alien Dropout
- Daredevil Dennis
- Snooker
- Diamond Mine
- Vortex

The conversion tape also allows you to configure most other games for joystick control.

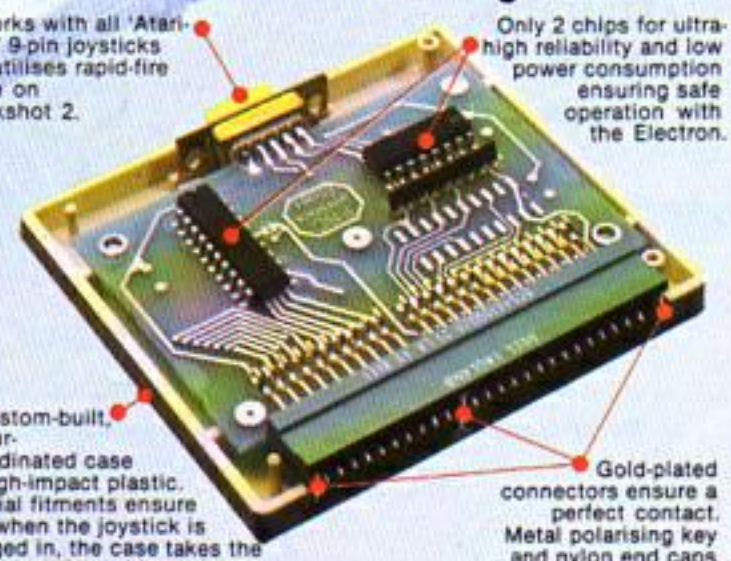
Games specially for the First Byte Interface

All these major software houses are bringing out games that work with the First Byte Electron Interface, with no conversion tape needed.

- Alligata
- A & F
- Program Power
- Superior
- Romik
- Bug-Byte
- Visions
- Virgin
- Aardvark
- Optima
- Postern
- Phoenix
- Software Invasion
- MRM
- Beebug-soft

The First Byte Electron Joystick Interface - available *now* from all good dealers and W. H. Smith.

Look at these advanced design features.



FIRST BYTE

A GENUINE FIRST BYTE ADD-ON

First Byte Computers,
10, Castlefields,
Main Centre, Derby.
DE1 2PE
Tel: Derby (0332) 365280

electron user NEWS

Jaguar guard for the star of the Show

A BIZARRE security operation was mounted on the eve of the Electron and BBC Micro User Show when an internationally famed gold artefact was transported across London.

The £30,000 work of art was moved from a city bank vault to Alexandra Palace around the neck of a snarling, 200 pound jaguar jungle wildcat.

Once safely there, it went on display during the four day major computer event.

"We had contacted a number of security firms to handle the job but all they could offer us were guards close to pensionable age and fleabitten alsations", says Mike Cowley, a spokesman for Database Publications, the show's organisers.

"And with the number of daring armed robberies taking place in London these days, we thought this might get it a one way ticket to a villa in Spain.

"So we decided on this unusual course of action. It was going to take a very daring thief to risk being turned into cat food by having a go".

Set with precious stones, the valuable bauble is known as the "Jewelled Hare of Masquerade".

It was the subject of a book called Masquerade published in 1979, and it subsequently became the object of an international treasure hunt undertaken by readers.

For "Masquerade" contained all the clues to find the hare which had been sealed in an earthenware jar and buried in a secret location by the author, Kit Williams, and television personality, Bamber Gascoigne.

When buried it was valued at £5,000. Three years later, when unearthed, its estimated worth had soared to more than £20,000.

Earlier this year, the precious item was bought by Haresoft Ltd, a London computer software company, to launch yet another hidden treasure competition with the hare as the prize.

This company is now marketing an electronic game in two parts – Hareraiser Prelude and Hareraiser Finale – which contain the clues to the eventual location of the hare. And the program has just become available for the Electron.

It was Haresoft who loaned the "Jewelled Hare" to Database Publications for the duration of the show.

PACE WINS THE DISC DRIVE RACE

THE July Electron and BBC Micro User Show saw the race to market the first Electron disc interface won by Pace of Bradford.

Named Le Box, the interface allows Electron users to use disc drives with their micro for the first time.

As well as speeding up the time taken loading and saving programs, Le Box owners will be able to create true random access files on their Electron. This means that the Electron has become a serious rival to the BBC Micro.

The interface is supplied in a self-contained unit with its own power supply plus a Pace 5¼in disc drive for £299.

Using the popular Amcom disc filing system and supplied with a comprehensive manual aimed at first-time users, Le Box will be available in mid August.

No Electron Graduate

RUMOURS that Data Technologies were planning an Electron version of their "Graduate" add-on for the BBC Micro have been dismissed by a company spokesman.

The Graduate virtually turns the BBC Micro into an IBM PC – and Data Technologies have had hundreds of orders for it.

"But there are no plans to bring out a Graduate for the Electron", said the spokesman.

Contract ties BBC to Acorn for four years

BBC Basic, the powerful language used on the Electron, has been given a huge boost with the announcement that Acorn Computers, have signed a contract with the BBC to continue to produce the BBC Micro for the next four years.

A blow to Sir Clive Sinclair's hopes of increasing his share of the educational market, this new lease of life for the BBC Micro also ensures that the Electron's structured Basic will be the educational standard for the forseeable future.

● Pictured signing the contract are (left to right) Acorn founders Hermann Hauser and Christopher Curry with Byron Parkin and Bill Cotton, managing director and chairman respectively of BBC Enterprises.

Whiz kids win road safety contest

A TEAM of computer whiz kids from the Holy Cross Convent School, New Malden, has won the Greater London road safety contest sponsored by *Electron User* and *The Micro User* magazines.

Open to all schools in the Greater London area, it was left to three fourth year girls from the school to come up with the best electronic answer to keeping death off the roads.

Karen Dyerson, Angela Moran and Sarah Finucane put their heads together and wrote the winning program.

As a result, they were invited to attend the opening day of the Electron and BBC Micro User Show held at Alexandra Palace to receive a disc drive as their prize from Database Publications, the show's organisers.

The winning program – tentatively entitled "Big Feet" – incorporated the slogan "When you Wanna Cross the Road Use the Green X Code".

"It was a first class example of how to use graphics to get the maximum impact", says Alan McLachlan, a road safety expert and chairman of the judging panel.

Adventures upgrade offer

ANY adventurer who is stuck in one of the first three adventures by Epic Software can now obtain an improved version free. Upgrades of Castle Frankenstein, Quest for the Holy Grail and Kingdom of Klein are available.

The new versions contain more clues and have a greater range of error messages. The amount of text has been increased, with an average of 25,000 characters in the Electron versions.

Epic has employed some of the compression techniques devised while writing Wheel of Fortune.

If you bought your adventure in the last three months you will probably already have the new version.

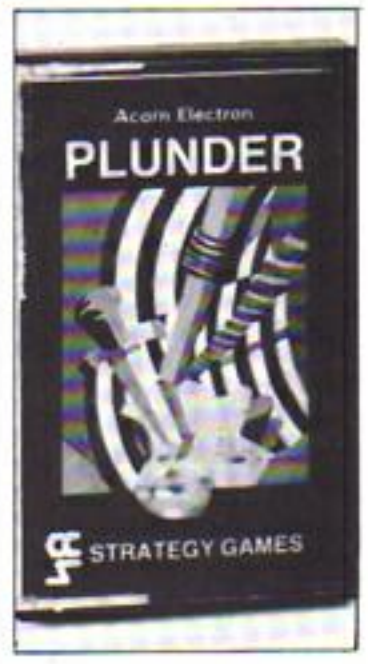
To check, simply load the first small Basic program and list it.

If the first line contains no version number, you have an early one and will be able to get a free replacement by returning it plus 50p P&P, to Epic Software, 10 Gladstone Street, Kibworth Beauchamp, Leicester LE8 0HL.

OLYMPICS SCORES

AFTER reaching No. 2 on the W.H. Smith software chart, Micro Olympics has jumped to the top BBC spot on the Micro Dealer Top 50.

Originally written for the Electron and BBC Micro, the 11-game package, which sells for only £5.95, has now been rewritten for the 16k Spectrum and Commodore 64.



PLUNDER, a strategy game of the Spanish Main, by Cases Computer Simulations, is now available for the Electron.

It creates the excitements of an Elizabethan sea captain in search of a knighthood and Spanish gold. It allows for either a short, 60 turns game or one twice as long, each of which can be played at three levels.

Software aid for games writers

ELECTRON users who would like to start writing games but who are not highly-skilled programmers can now get help from software.

The authors, Yorkshire-based Holly Computers, say you still need to know a little Basic to use their Gamemaker 2 package.

It is also handy if you have started to write your own games.

According to Holly the package bridges the skill gap and allows Basic programmers to convert their games ideas into action.

Gamemaker takes the idea of sprites – user defined multi image and multi coloured characters of various sizes – and advances it with the addition of simple commands inserted in the Basic program.

Each sprite has two images assigned to it, which are drawn on the screen alternately. By designing slightly different images you create the impression of animation.

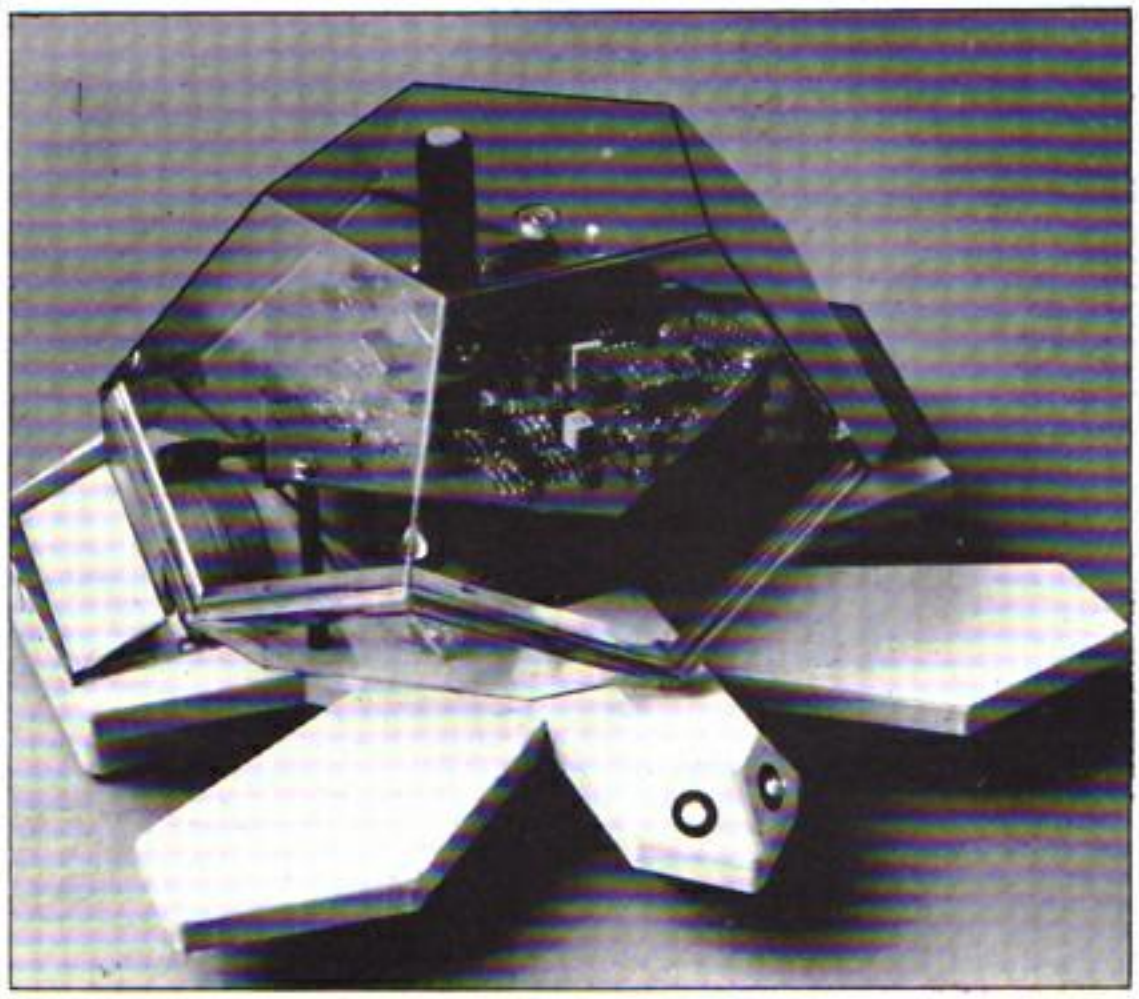
The product consists of an image designer program, sprite control routines, a 40 page user guide and a demonstration program.

ON TAP FOR ELECTRON

SIX popular BBC Micro titles have been converted for the Electron by Acornsoft.

The new conversions include three arcade-type games, Hopper, Freefall and Arcadians, as well as an adventure game, Sphinx Adventure, and two home interest programs, Desk Diary and Picture Maker.

All are on cassette at £9.20.



Go-it-alone turtle..

THIS is claimed to be the world's first remote controlled turtle, from Valiant Designs. Aimed at junior schools and hobbyists, it can be used with popular micros, including the BBC Micro, Apple and – shortly – the Electron.

While young children program the turtle on the keyboard to walk, dance and play games, older ones can use it to draw complex designs using the Logo language.

Accent on learning

TO follow their foreign language learning aids, The French Mistress, The German Master and The Spanish Tutor, Kosmos Software has launched the first in a new series of educational programs for the BBC and Electron Micros.

Aptly titled Answer Back, the first release covers general knowledge for the over-elevens.

It combines quizzes on general topics with a space age game.

Fifteen quizzes are supplied on a cassette. Subjects include science, history, geography, astronomy, music and sport.

And a master control program enables unlimited new quizzes to be created.

No holds barred soccer

THE cloistered peace of a respected academic institution was shattered recently when students nearly came to blows over a computer game.

They were playing Qual-Soft's soccer strategy game, "League Division One", as part of a computer appreciation course.

But these students also happened to be apprentices from two rival football clubs, Swindon Town and Reading, on day release from Swindon College.

Passions ran high as the students argued the best strategy for the game, which simulates the problems of a football manager as he steers his team to the championship.

League Division One is available for the BBC Micro, and an Electron version is also due soon.

Command performance

THE Queen saw the Electron in action when she opened the Women's Institute "Life and Leisure" exhibition at Olympia.

Acorn, the only computer firm at the show, set up 29 Electrons and BBC Micros to teach visitors what the micro could do in the home, at school and for the home-run business.

In line with the theme of the show nearly all Acorn's stand demonstrators were women.

Acorn's stand manager, Rachel Pullen, reported: "We had more than 100 visitors on the stand at once for much of the time.

"For many, it was the first time they had used a computer".

Acorn is currently trying to raise women's interest in computers.

Last year they published a survey which indicated that girls were 13 times less likely than boys to be using a home computer.

And they say they are now determined to do something about it.

Part eight of PETE BIBBY's introduction to programming

Build a nest in your micro

More about FOR . . . NEXT loops

LAST time we took another look at FOR . . . NEXT loops and introduced the command STEP which allowed us to vary the amount that the control variable changed each time round the loop.

This time we'll be taking another look at FOR . . . NEXT loops but concentrating on what are known as nested loops.

What's a nested loop? Well, take a look at Program I, which is the previous article's Program X:

```
  10 REM PROGRAM I
  20 REM OLD PROGRAM X
  30 FOR  outer=1 TO 3
  40 PRINT "Outer loop nu
mber ";outer
  50 FOR inner=1 TO 3
  60 PRINT "Inner loop ";i
nner
  70 NEXT inner
  80 NEXT outer
```

This produces – or should if you've typed it in properly – the following output:

```
Outer loop number 1
Inner loop 1
Inner loop 2
Inner loop 3
Outer loop number 2
Inner loop 1
Inner loop 2
Inner loop 3
Outer loop number 3
Inner loop 1
Inner loop 2
Inner loop 3
```

Not very exciting, but it does show nested loops in action. There's obviously an outer loop with the control variable *outer* and an inner loop controlled by *inner*. But how do they work?

To answer that let's take a look at Program II, a very simple listing with one FOR . . . NEXT loop in it:

```
10 REM PROG II
20 FOR asterisk=1 TO 10
30 PRINT "*";
40 NEXT asterisk
50 PRINT
```

Elegant, isn't it? All it does is produce:

****

The loop cycles 10 times printing out an asterisk each time it does so. The semi-colon ensures that they all appear on the same line.

If you don't understand how the program works you'd better re-read the articles in the July and August issues of *Electron User* to refresh your memory of FOR . . . NEXT loops.

Now suppose you wanted – for reasons best known to yourself – five rows of 10 asterisks. You could get them by writing a program that uses the FOR . . . NEXT loop of Program II five times in succession.

However there's a much better way of doing it as shown in Program III. Run it and see what happens. Exciting, isn't it?

Some of the program is familiar. Lines 30, 40 and 50 are taken directly from the previous program. The FOR . . . NEXT loop formed by these lines obviously produces the rows of ten asterisks.

```
10 REM PROG III
20 FOR row=1 TO 5
30 FOR asterisk=1 TO 10
40 PRINT "*";
50 NEXT asterisk
55 PRINT
60 NEXT row
70 PRINT
```

But how do we get five of these rows?

The answer is that we've put the FOR . . . NEXT loop we took from Program II inside another FOR . . . NEXT loop formed by lines 20 and 60.

This outer FOR . . . NEXT loop cycles five times as the control variable *row* changes in value from 1 to 5.

Each time round the loop the program obeys the lines that it finds between the FOR of line 20 and the NEXT of line 60.

Now these lines themselves happen to form an inner FOR . . . NEXT loop with the control variable *asterisk*. When it comes to this loop the Electron obeys it in exactly the same way as it did in Program II, producing the required line of ten asterisks.

It then comes to the NEXT of line 60 and, unless the control variable *row* is greater than five, the program goes back to the FOR of line 20 and begins the whole thing over again.

Hence the five rows of ten asterisks.

```
**********
**********
**********
**********
**********
```

To put it another way, there is an outer loop (with control variable *row)* and an inner loop (with control variable *asterisk).* The outer loop is repeated five times.

The first time round the outer loop, *row* is equal to 1. The program then meets the inner loop and obeys that, the control variable *asterisk* going through all its values from 1 to 10.

This produces one line of asterisks.

Now the inner loop is finished the program carries onto the next lines and meets the NEXT of line 60.

Since *row* is only 1, the outer loop is still operating. The outer loop control variable *row* now has one added to it, becoming 2.

The program carries on and meets the inner loop, again obeying it 10 times, with the resultant row of asterisks.

When the inner loop has finished its work the program carries on, finds that it hasn't satisfied the conditions of the outer loop and so performs it once more – *row* now being 3.

It only finishes when the outer loop has been performed five times, giving the five rows of 10 asterisks.

Program IV shows how the outer loop cycles five times, with the inner loop going through all 10 of its cycles each time.

```
  10 REM PROG IV
  20 FOR row=1 TO 5
  30 PRINT"Outer loop cycl
e ";row
  40 FOR asterisk=1 TO 10
  50 PRINT "*";
  60 NEXT asterisk
  70 PRINT
  80 NEXT row
  90 PRINT
```

Notice how in line 30 I have used the control variable *row* to give the number of that cycle.

The PRINT commands of lines 70 and 90 just make things neater. Leave them out and see what happens.

```
Outer loop cycle 1
**********
Outer loop cycle 2
**********
Outer loop cycle 3
**********
Outer loop cycle 4
**********
Outer loop cycle 5
**********
```

Now that we've seen how we can have one FOR ... NEXT loop nested inside another, Program I should make a lot more sense.

Let's see what we can do with them. Have a go at Program V:

```
10 REM PROG V
20 FOR row=1 TO 5
40 FOR asterisk=1 TO row
50 PRINT "*";
60 NEXT asterisk
70 PRINT
80 NEXT row
90 PRINT
```

This is very much the same as the previous programs. There's an outer loop with a control variable *row* going from 1 to 5.

The inner loop is slightly different. It still has the control variable *asterisk* but now *asterisk* has the range from 1 to *row*.

Run the program and you'll see that you get a triangle made of asterisks:

```
*
**
***
****
*****
```

If you manage to control your excitement it's easy to see what's happened. The outer loop behaves just as before with *row* being 1, then 2, and so on to 5.

The inner loop goes through all its cycles for each pass round the outer loop, just as before.

What's different is that the inner loop's control variable goes through the range 1 to *row* each time and not 1 to 10 as before.

The number of times that the inner loop is performed – and hence the number of asterisks printed on that line – depends on the value of *row*.

The first time round the outer loop *row* is 1 and so the inner loop prints only 1 asterisk.

The second time round the outer loop, *row* has the value 2. This means that the control variable of the inner loop, *asterisk*, now ranges from 1 to 2. The inner loop cycles twice and we get two asterisks.

When *row* is 3 the inner loop cycles three times with three asterisks being printed on that line, and so it carries on.

If you can't understand it, try working through the program line by line, noting down the values of *row* and *asterisk* at each step. You'll soon see what happens.

Remember that the inside loop goes through all its cycles each time round the outer loop. All we've done in Program V is to make the number of cycles of the inner loop dependent on the how many times the outer loop has been performed.

Try Program VI out:

```
10 REM PROG VI
20 FOR row=5 TO 1 STEP -
1
40 FOR asterisk=1 TO row
50 PRINT "*";
60 NEXT asterisk
70 PRINT
80 NEXT row
90 PRINT
```

This is more or less the same as the previous program except that I've used a STEP of –1 to make *row* go down from 5 to 1. This means our triangle of asterisks is now upside down:

```
*****
****
***
**
*
```

The previous two programs used the control variable of the outer loop to influence the inner one. It's very often the case that the control variables we use to govern our loops are also used inside them.

Take a look at program VII, which produces a quadrilateral of asterisks:

```
10 REM PROGRAM VII
20 MODE 6
30 FOR column=0 TO 19
40 FOR row= 0 TO 19
50 PRINT TAB(column, row
)"*"
60 NEXT row
70 NEXT column
```

Here we have a pair of nested loops. The outer loop has the control variable *column* which varies from 0 to 19. The inner loop has *row* as its control variable.

The interesting part is line 50 which uses the values of *row* and *column* with TAB to position the asterisks.

Each time round the outer loop the inner loop cycles 20 times, producing a column of asterisks. The next time round the column is one space to the right and so on.

Program VIII does exactly the same thing but in a different way:

```
10 REM PROGRAM VIII
20 MODE 6
30 FOR row= 0 TO 19
40 FOR column=0 TO 19
50 PRINT TAB(column, row
)"*"
60 NEXT column
70 NEXT row
```

Now the output appears row by row. This is because I've made the inner loop of Program VII the outer loop of Program VIII.

And that's about all for this month. We've seen how loops can be nested inside each other and also how the control variable of the outer can effect the inner loop.

We've also seen how the loop control variables can be used inside the loops, in this case deciding the position of asterisks.

Until next time I'll leave you with this problem. Both the triangles of asterisks we produced had their hypotenuses to the right.

Can you modify the programs to produce triangles of asterisks with the hypotenuse to the left as in Figures I and II?

```
    *
   **
  ***
 ****
*****
```

Figure I

```
*****
 ****
  ***
   **
    *
```

Figure II

DAVE ROBINSON'S

MATHS workout

Exercises for the Electron

BY now you're probably familiar with numeric variables – those used to store numbers – and you also know that there are two types, real and integer.

Real variables are used to store whole numbers and numbers having a decimal part – called real numbers. Examples are *count* =150 or *weight* =14.75.

Integer variables can only be used to store whole numbers and must be distinguished by the percentage sign such as *lives%* =2 and *column%* =5.

Both real and integer variables are used in programs but there is no doubt that, where possible, integer variables are preferable.

This is because they take up less memory space – four bytes against five for real variables – and allow a program to run faster. Integer variables are also more accurate than real ones.

Having said all that in favour of integers, there can be times when strange effects can be obtained from the simplest of programs which use integer variables. This is when division arithmetic is used.

Take a look at Program I.

```
  10 REM PROGRAM I
  20 litresofwine=10
  30 people=20
  40 share=litresofwine/pe
ople
  50 PRINT share" litres"
```

Program I

This program is easy to follow and uses real variables throughout. If you type it in you should get the answer you'd expect.

Now look at Program II. It's the same program, but this time it uses integer variables.

```
  10 REM PROGRAM II
  20 litresofwine%=10
  30 people%=20
  40 share%=litresofwine%/
people%
  50 PRINT share%;" litres
"
```

Program II

When you type and run Program II you'll wonder where the wine went to. (The EEC wine lake perhaps?)

The problem occurred on line 40 when the computer attempted to make the variables *share%* equal to 0.5, which is not an integer.

The actual number stored in *share%* was not 10/20 but 10 DIV 20. DIV is the Electron's shorthand for integer division.

The result of a DIV operation is the whole number part of the answer. If there are any fractions or decimal parts to the answer they are just ignored.

There is no rounding up when you use DIV. You just get the whole number part of the answer. Anything else is completely lost.

Going back to Program II, we should have left *share* as a real variable with a line like:

```
share=litresofwine%/people
%
```

The trouble with DIV is that it throws away the fractional part of the answer.

However there is a method to find the other part of an integer division. It's the Electron's other integer operator, MOD. No, it's not back to the swinging sixties but MOD as in MODULO.

If you MOD an integer number you're still dividing. But now the answer you get isn't the whole number from the division but the remainder.

Remainders were the numbers left over in division sums in our early school days, before we got clever and learnt fractions and decimals. Yes it's that simple! What you get from MOD is the bit that DIV discards.

Let's look at some examples, first in ordinary arithmetic:

7 divided by 3 = 2 remainder 1
19 divided by 4 = 4 remainder 3
9 divided by 5 = 1 remainder 4.

In 'MOD & DIV' arithmetic this becomes:

7 DIV 3 = 2 and
7 MOD 3 = 1
19 DIV 4 = 4 and
19 MOD 4 = 3
9 DIV 5 = 1 and
9 MOD 5 = 4

To understand this more fully type in Program III and run it several times. You can then see the results you get from integer division using both MOD and DIV.

```
  10 REM PROGRAM III
  20 INPUT "Type a whole n
o."N1%
  30 INPUT "Type another n
o."N2%
  40 PRINT N1%;" DIV ";N2%
;"=";
  50 PRINT ;N1% DIV N2%
  60 PRINT N1%;" MOD ";N2%
;"=";
  70 PRINT;N1% MOD N2%
```

Program III

Start by keeping your numbers small and try to predict the answers the computer gives before they come up on the screen.

You can try to confuse the computer by typing in real numbers (with a decimal part). You'll not find any problems, the Electron's too clever for that.

What it does is to truncate – cut off – all the decimal part of any number when using integer variables. It also does this when the number follows the MOD and DIV operator, so 9.3 DIV 3.65 becomes 9 DIV 3.

If you run Program III often enough you may notice that the result from a MOD division is never more than the number itself.

The result will always be between zero and one less than the MOD number used. *X*% MOD 9 will give an answer between 0 and 9 regardless of the value of *X*%.

This fact can be used when writing our programs, but before I show you how, we can check this phenomenon by using the Electron itself.

When you type in a number after a command such as COLOUR or MODE then, before obeying the command, the Electron does a quick MOD on it to make it lie within the required range.

Select MODE 5 and type COLOUR 1. A red cursor will appear on the next line on the screen.

Now type COLOUR 5. The cursor will stay red because in a four colour mode, the Electron applies a MOD 4 to the number so only colours 0, 1, 2, and 3 are accepted.

By now you will know the answer to 5 MOD 4. Yes, you're right it's 1.

Still not convinced? Try COLOUR 10. It should be yellow as 10 MOD 4 is equal to 2 and COLOUR 2, in MODE 5, is yellow.

If you were to type a real number after the COLOUR command, the micro truncates it as before, then acts as if it was an integer after all.

The MODE command is similar except, being the baby

brother of the BBC Micro, the Electron does a MOD 8 on the number, not a MOD 7, as you may have expected.

This means that if you type MODE 15 the Electron does a MOD 8 and gets MODE 7. Then it internally switches to MODE 6. Don't worry, it's all in aid of compatibility.

Now, back to programming. If your programs offer a selection to the user – level of play perhaps – then you will need to incorporate a routine to check the input against what the program has been designed to use.

This is essential because, without a check, bad input will probably cause the program to crash. Using the knowledge gained from the Electron we can easily prevent this happening. Look at Program IV.

```
  10 REM PROGRAM IV
  20 INPUT "Type a number
(0-4)" N%
  30 input%=N% MOD 5
  40 PRINT input%
```

Program IV

Program IV will ensure that *input%* becomes an integer between 0 and 4 regardless of what was typed in. To check this for yourself run it and experiment with different numbers.

Unfortunately it is still possible to crash this listing, but this will only happen if the number you type in is too large for the Electron to handle.

The highest number the Electron can store in a four byte integer is:

2,147,483,647

If in Program IV you type a number greater than this, the program will crash with the message: 'To big at line 20'. This problem will apply to all your programs that use integer numbers.

The final program shows the use of both MOD and DIV to convert a whole number of minutes into minutes and hours.


```
  10 REM PROGRAM V
  20 INPUT "Total no. of m
inutes" M%
  30 hours%=M% DIV 60
  40 min%= M% MOD 60
  50 PRINT;hours%;" HOURS
and "min%;" MINUTES"
```

Program V

If you wish to type higher numbers then add the lines:

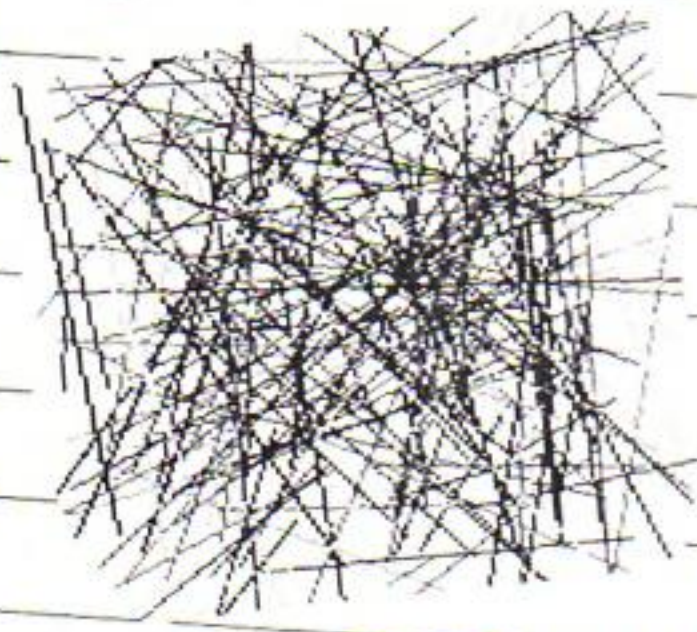
```
  24 days%=M% DIV 1440
  26 M%=M% MOD 1440
  45 PRINT ;days%;" DAYS "
;
```

Why 1440? That's how many minutes there are in a day!

Notebook Part 8

THIS month's Notebook program comes from IAN RODGERS of Grantham. It's called Wild Computer – and when you run it you'll know why! Apart from being an entertaining program, it's also a nice illustration of nested loops.

Send your micro absolutely WILD!



10,20 REMs to tell us the program name and who wrote it. No marks for guessing that this line puts the Electron into Mode 2.

30 Switches off the flashing cursor.

40 These lines form the outer loop of a set of two nested loops. The loop control variable *A* decreases from 255 to 1 in steps of 20.

50,140 These lines form the boundaries of the inner loop of the two nested loops. The loop control variable *Z* increases from 1 to 255 in steps of 20. Notice that the inner loop works through all its cycles for each time that the outer loop cycles once.

60,130 This is where the noise comes from. Line 70 makes the same noise each time the program executes it, while the pitch of the notes SOUNDed by the next two lines depends on the values of *A* and *Z* at that time. The pitch of the noise depends on the value of *A*, the outer loop variable. This will be decreasing as the program goes on.

70-90 The pitch of this note is given by the value of the inner loop control variable *Z*. This is increasing from 1 TO 255 and so gives a note that is rising in pitch. Notice that the nesting of the loops means that the whole range of *Z* is covered for every value of *A*.

80 The randomly coloured lines are drawn by this part of the program.

90 These lines just hold up the program until a key is pressed when the whole thing is RUN again.

100-120

150,160

```
10 REM WILD COMPUTER
20 REM BY IAN RODGERS
30 MODE2
40 VDU23,1,0;0;0;0;
50 FOR A=255 TO 1 STEP -20
60 FOR Z=1 TO 255 STEP 20
70 SOUND 1,-15,5,1
80 SOUND 1,-15,A,1
90 SOUND 1,-15,Z,1
100 MOVE RND(1279),RND(1023)
110 GCOL0,RND(16)
120 DRAW A*4,Z*4
130 NEXT
140 NEXT
150 A$=GET$
160 RUN
```

Outer loop

Inner loop

These make the noises

These draw the lines

Holds things up until a key is pressed

Trevor Roberts

NEW
ELECTRON INTERFACE UNIT

Expand your Electron.

Now you can use your Electron computer with any standard printer using MUSHROOM's new printer and user-port interface.

Bring your Electron up to the same standard as the BBC Model B computer printer and user-port into which you can plug robot arms, joy sticks or any BBC user-port module.

On the whole range of MUSHROOM modules and interfaces, the Electron edge connector is extended to give you unrestricted compatibility with any other Electron interface.

All MUSHROOM interfaces can be used separately or can be combined into the unique MUSHROOM ELECTRO-RAK which is conveniently connected to the Acorn Electron by a short cable.

This greatly enhances the performance of the Electron and turns the system into a computer comparable with many larger mini systems and as you grow your computer can grow with you!

Ask for details on:

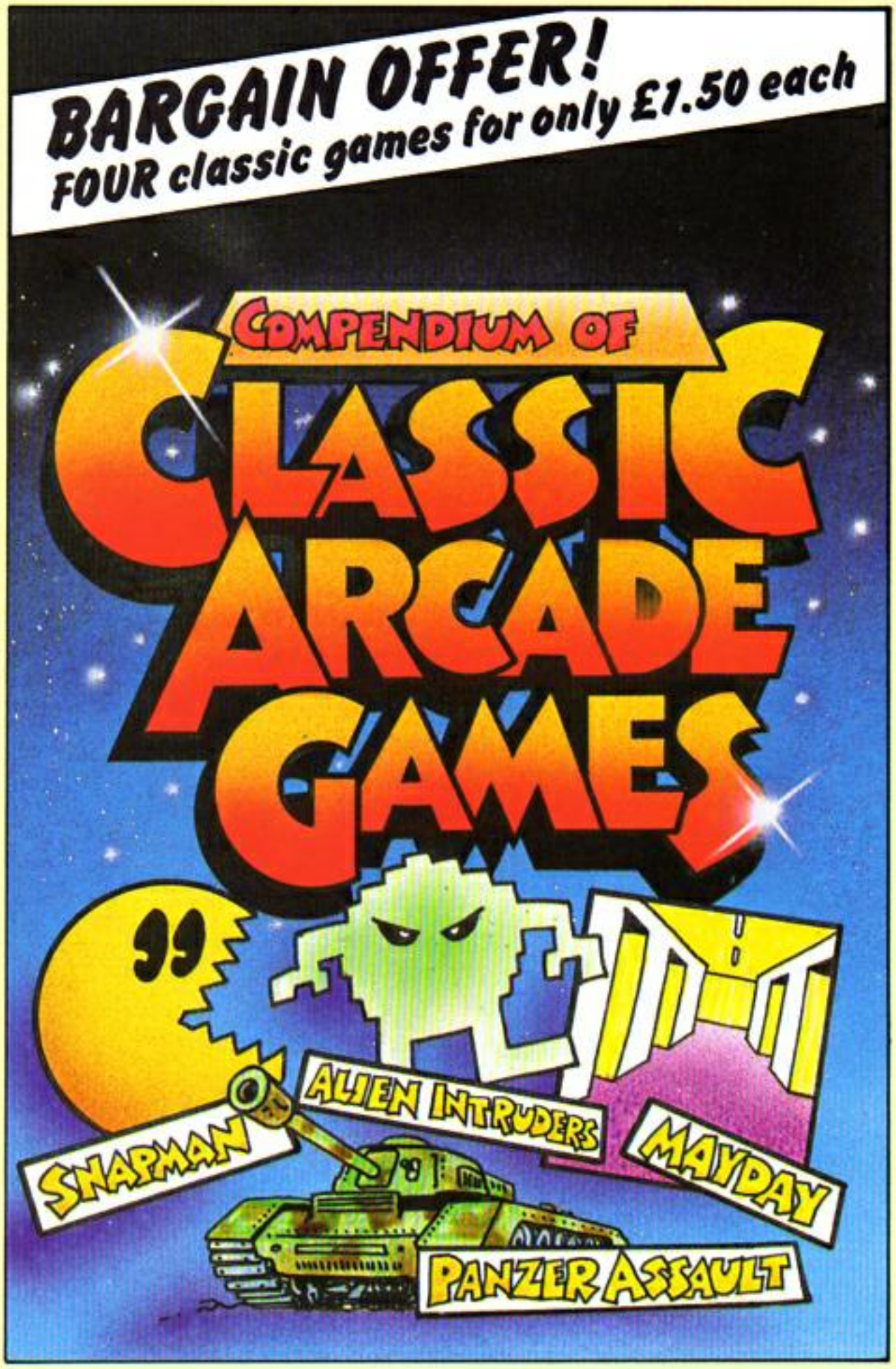
* SIDEWAYS ROM CARD
* A-D JOYSTICK INTERFACE
* EPROM PROGRAMMER
* MUSHROOM ELECTRO-RAK

* FULLY CENTRONICS COMPATIBLE
* FULL SOFTWARE PACKAGE (including screen-dump routine)
* BBC COMPATIBLE 8-BIT USER PORT

£39.95 Ex VAT

Mushroom
COMPUTERS LIMITED Aston Road, Bedford, Beds MK42 0LJ. **Telephone: (0234) 58303.**
Another Mushroom product from Broadway Electronics.

Here's something SPECIAL from



electron user

We've commissioned four rip-roaring games for the Electron and BBC Micro

Three of this high-powered collection are top-rate machine-code versions of arcade classics and the fourth is a thrilling real-time adventure game. There's hours of enjoyment and something to suit everyone in this unique value for money collection

SNAPMAN – Guide your man through the maze as he munches energy pellets and avoids hostile aliens

ALIEN INTRUDERS – With only your laser for protection you must destroy the waves of aliens who threaten to engulf you

PANZER ATTACK – You are a tank commander engaged in vicious combat against encircling enemy forces

MAYDAY – A futuristic adventure! As captain of an interstellar cruiser you must guide the sole survivor of a stricken space freighter through the wreckage of his craft. If you fail to recover those vital medical supplies a whole planet is doomed!

Please send _____ copy/copies of Classic Arcade Games.
I enclose a cheque/PO No. _____
for £ _____
made payable to: Database Publications Ltd.

- ☐ Electron tape £5.95
- ☐ BBC Micro tape £5.95
- ☐ BBC Micro disc £7.95

(Please tick)

Name _____

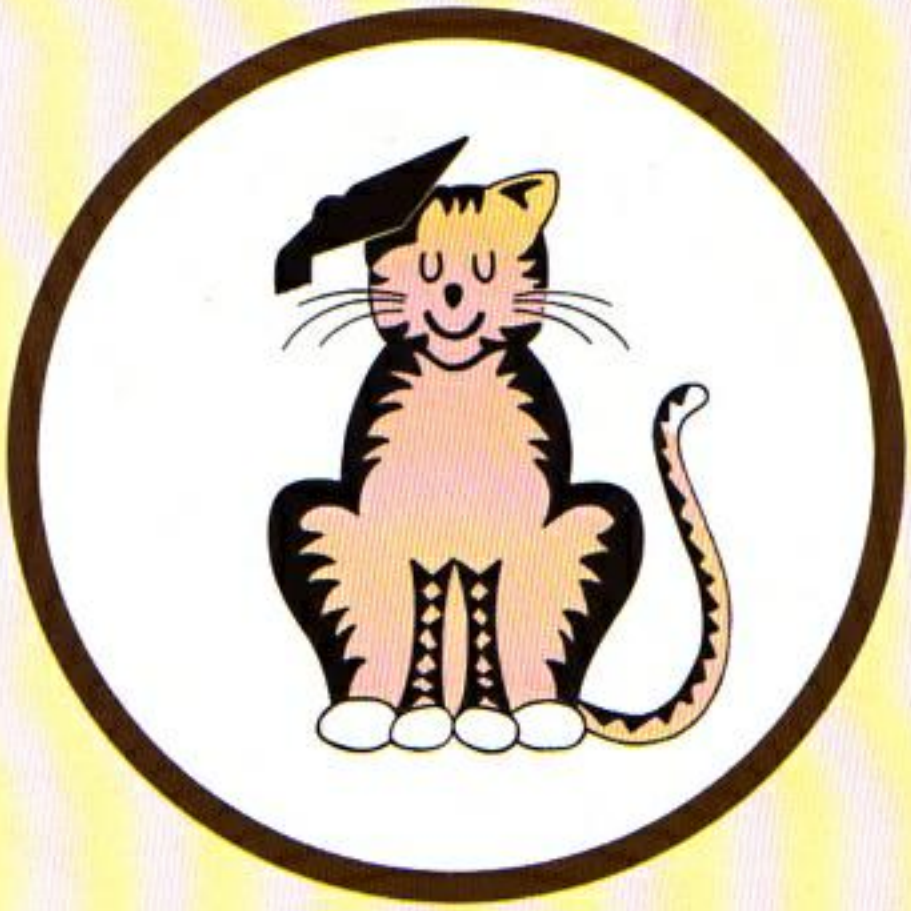
Address _____

_____

Post code _____ Tel: No. _____

Post to: Classic Arcade Games offer, Electron User, 68 Chester Road, Hazel Grove, Stockport SK7 5NY

CHESHIRE CAT
EDUCATIONAL SERIES
from
AMPALSOFT



CHESHIRE CAT

The First name in Educational Software.

An exciting range of top quality programs

Available for BBC, Electron, Dragon 64, Dragon 32.
Available shortly for Commodore 64, Spectrum 48K.

RETAILER ENQUIRIES WELCOME

Ampal Computer Services Ltd.,
31 Woodbridge Road, Darby Green, Blackwater,
Camberley, Surrey.
Tel: (0252) 876677.

Available from selected branches of Boots, W. H. Smith and Dragon Data Ltd., and other good computer stores

DOCTOR SOFT
ADVANCED SOFTWARE

DOCTOR SOFT
P.O. BOX 66
EAST PRESTON
WEST SUSSEX

NOW AVAILABLE FOR THE ACORN ELECTRON £7.95

747
Cockpit view, pilot written, instrument & visual 747 Flight Simulator. Banking & pitching 3D outside view of Horizon/Runway (Heathrow/Gatwick). 7 Nav points with continuous Navigation computation. Joysticks or Keyboard option, briefing program, map, notes & flight plan. Demonstration approach, 4 colour — mode graphics & sound.
Only £8.95 inc. VAT & PP (Disc £11.95)

GORPH
(PURE MACHINE CODE, ARCADE QUALITY)
The first BBC version of this superb machine code Arcade favourite! 4 widely different screens of high speed action: Invaders, Laser attack, Firebird, & Mothership, all in smooth 16 colour-mode graphics!
Only £7.95 inc VAT & PP (Disc £10.95)

MISSILE ATTACK
(PURE MACHINE CODE, ARCADE QUALITY)
Another well known arcade favourite, 2 player/ Joystick options, remote target designation of incoming ballistic missiles & attack craft. Protect your cities! Incredible Armageddon graphics & sound! *£7.95*

DOUBLE ACTS
2 GAMES, 1 THEME,
Amazing value at only £6.95 per pair:

SPOOK, SPOOK:
MUNCHER 2 ghosts, 20 mazes, 3 skill levels, hall of fame GHOST MINE Dig for gold, watch for spooks & snakes.

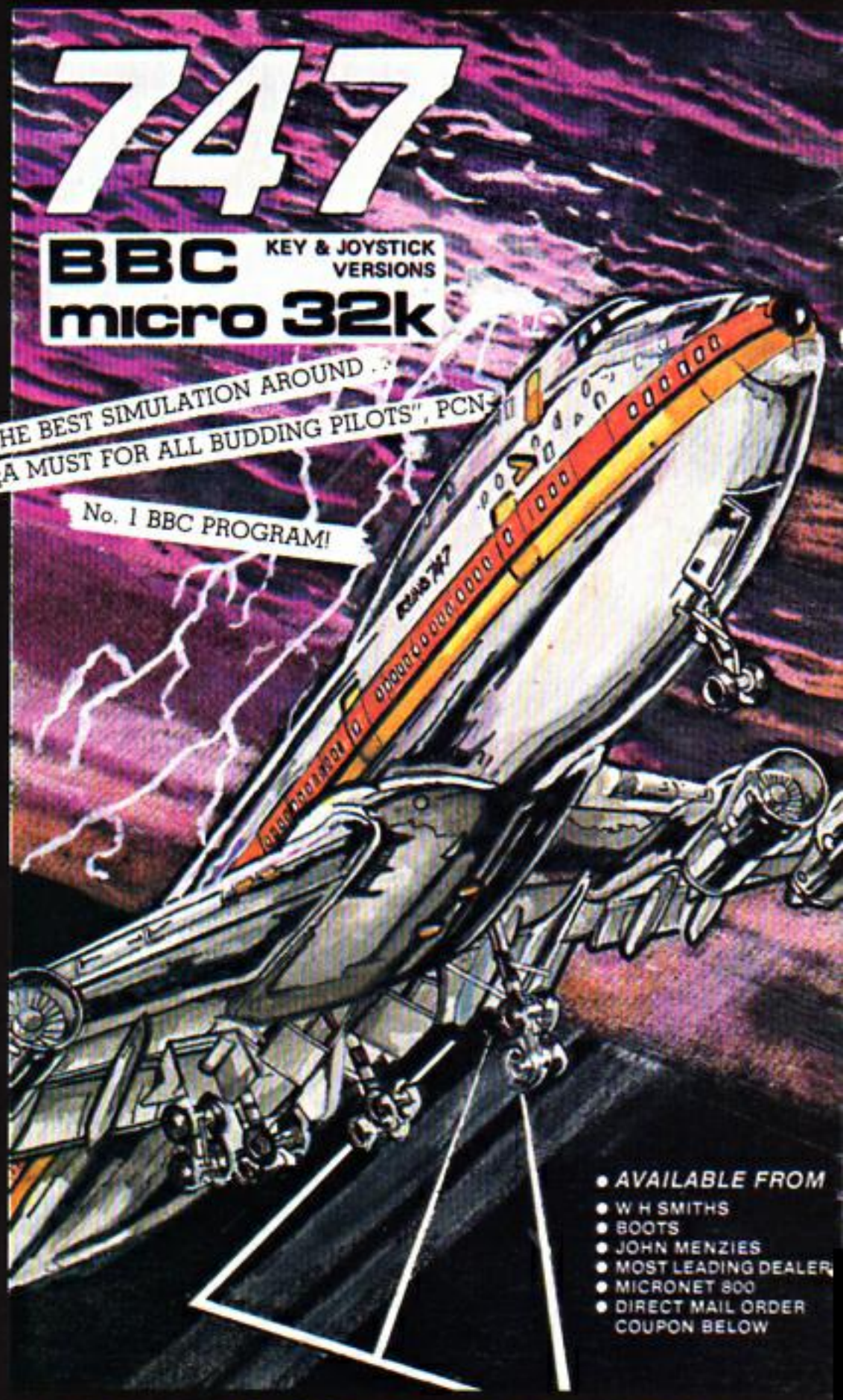
SPACE, SPACE:
WOLFPACK 3 starships, 1 space station, 4 galaxies, unlimited motion in space! MISSION ALPHA 3D High speed action, hall of fame & music!

SPY, SPY:
KREMLIN Escape through the endless 3D corridors of the Kremlin, aided by map & compass, but watch for the Gremlins!
BONDSKI Lethal action as James skies down the slope & parachutes into the void!

WORD PROCESSOR:
WORD PERFECT £8.95 cassette £11.95 Disc
Full facility 40/80 column word processor, wrap around, block move, justification, word replace, etc. With full instructions & Key insert.

ONE DISK HOME OFFICE:
Complete Menu driven home office system including WORD PERFECT, WORDKIT, LETTER, CALCULATE (mini spread-sheet), & DIRECTORY (expandable card file system). Supplied with detailed manual. We believe this to be the best value package of its kind ever offered.
Only £19.95 Disc

EDUCATIONAL SOFTWARE:
TALKING TABLES TEACHER (7-14yr old) Speaks when Speech ROM fitted, teaches multiplication with colourful games & questions. *£6.95*
COLOUR SHAPE MATCH (2-6yr old) Beautiful suit of 3 programs teaching shape & colour recognition with delightful graphics & sound. *£6.95*



DEALERS CONTACT 0902 70044

MAIL ORDER

ALL PRICES INCLUDE VAT, POSTAGE + PACKING FREE

TITLES .....

.....

QUANTITY .....

.....

DISK ☐ CASSETTE ☐

I ENCLOSE CHEQUE/PO, ACCESS NO., VALUE .....

NAME .....

ADDRESS .....

.....

WE WELCOME HIGH QUALITY PROGRAM SUBMISSIONS AND PAY TOP ROYALITY RATES

Now YOU can go for gold ...with the MICRO OLYMPICS

Fancy pitting yourself against the world's best at this summer's Olympics?

You can do so without going anywhere near Los Angeles – with the most challenging package of programs of 1984.

MICRO OLYMPICS is more than a game. It's a brilliantly written collection of ELEVEN track and field events.

And because we know we're going to sell many thousands of them we've brought the price right down – to just £5.95.

Ever imagined yourself as another Seb Coe? Then try to run against the world record holder at 1500 metres. And if that distance is too much for you then there's always the 100, 200, 400 and 800 metres to have a go at.

Not much good at running? Don't worry, MICRO OLYMPICS has many more challenges for you. Why not try your skill at the high jump or the long jump?

And if you can't beat the computer at running or jumping then you can always throw things around in frustration! The trouble is that it's just as hard to be a champion at the discus, the hammer or the javelin.

And the pole vault takes the event to new heights!

Yes, it's fast, furious fun, pitting yourself against the world's best times and distances on your micro.

You may not be another Steve Ovett or Alan Wells, but with practice you COULD become the Micro Olympics Champion!

Also available from WH Smith and all other leading stores

Play Micro Olympics – and let your fingers do the running!

Send for it today

Please send me _____ copy/copies of Micro Olympics

☐ I enclose cheque made payable to Database Publications Ltd. for £ _____

☐ BBC 'B' cassette	£5.95
☐ Electron cassette	£5.95
☐ BBC 40-track disc	£7.95
☐ BBC 80-track disc	£7.95

Please tick box

I wish to pay by
☐ Access ☐ Visa No. _____ Expiry date _____

Signed _____

Name _____

Address _____

_____

Post to: Micro Olympics offer, Database Publications, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

SCRAPBOOK

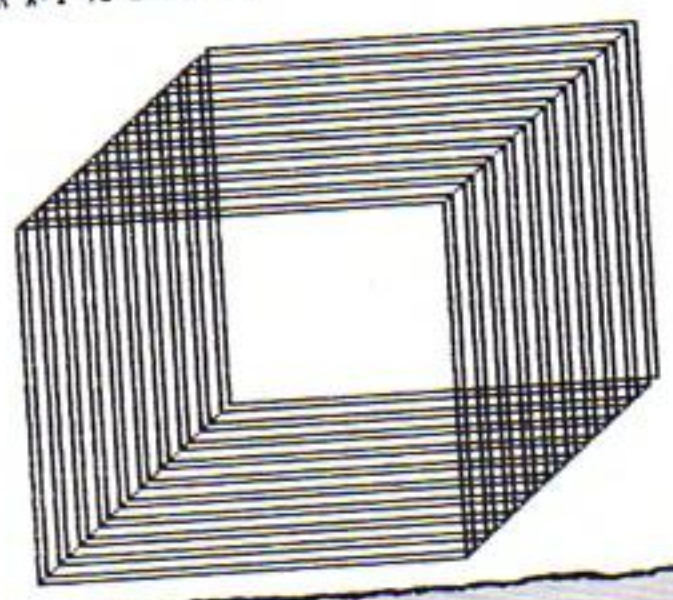
SCRAPBOOK is a new feature consisting of short, simple programs sent in by our readers. It's where we keep a record of all the interesting little routines that don't end up in our regular Notebook or Program Probe features but are too good for us not to share with other users.

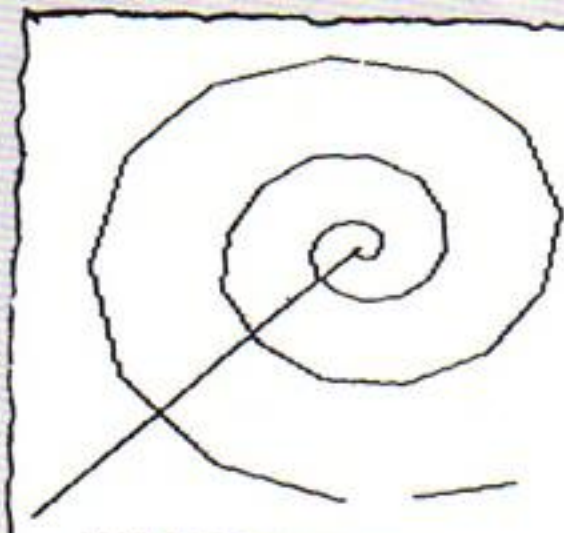
This month the emphasis is on graphics programs. Next time it will be on . . . well, that's up to you because Scrapbook is made up of the programs you send in. So graphics programs, utilities, maths programs and simple games – they're all welcome here. Keep them coming and see yourself in print!

3D CUBE

The ins and outs of perspective in this program from Jason Fox.

```
10 REM 3D CUBE
20 REM JASON FOX
30 REM HAVERHILL,SUFFOLK
40 MODE 1
50 VDU23,1,0;0;0;0;
60 PRINTTAB(5,10)"This prog
ram draws a 3D cube."
70 PRINTTAB(15,12)"Press sp
ace."
80 WAIT$=GET$
90 CLS
100 A=200:B=600
110 FOR N=1 TO 20
120 PROCCUBE(A,B)
130 A=A+10:B=B+10
140 NEXT
150 MOVE 200,200:DRAW 380,38
0
160 MOVE 200,600:DRAW 380,78
0
170 MOVE 600,600:DRAW 780,78
0
180 MOVE 600,200:DRAW 780,38
0
190 FOR X=1 TO 2000:NEXT
200 CLS
210 PRINTTAB(2,10)"To see a
view of the cube from inside"
220 PRINTTAB(15,12)"press sp
ace."
230 WAIT$=GET$
240 CLS
250 A=300:B=700
260 FOR N=1 TO 20
270 PROCCUBE(A,B)
280 A=A-10:B=B+10
290 NEXT
300 MOVE 300,300:DRAW 120,12
0
310 MOVE 300,700:DRAW 120,88
0
320 MOVE 700,700:DRAW 880,88
0
330 MOVE 700,300:DRAW 880,12
0
340 END
350 DEF PROCCUBE(Z,X)
360 GCOL 0,1
370 MOVE Z,Z
380 DRAW Z,X
390 DRAW X,X
400 DRAW X,Z
410 DRAW Z,Z
420 ENDPROC
```

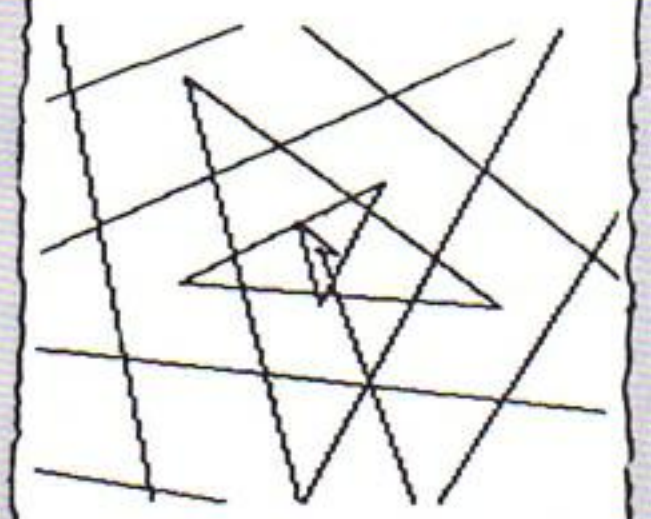


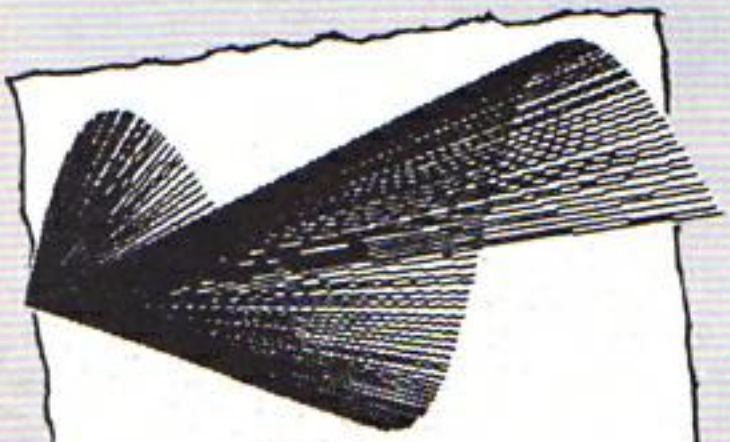


LINE DRAWING

Basic and trig combine in Stephen Martin's doodle program.

```
10 REM LINE DRAWING
20 REM STEPHEN MARTIN
30 REM NORWOOD,LONDON
40 XC=0:YC=0
50 Q=0.5
60 MODE 2
70 VDU29,612;540;
80 VDU 23,1,0;0;0;0;
90 FOR T=1 TO 50 STEP Q
100 X=COS(T)
110 Y=SIN(T)
120 XC=X*T:YC=Y*T
130 DRAW XC*T,YC*T
140 NEXT
150 Q=Q+0.5
160 CLS:GOTO 90
```





WAVES

More trig patterns from Stephen Martin.

```
10 REM WAVES
20 REM STEPHEN MARTIN
30 REM NORWOOD,LONDON
40 MODE 4
50 VDU23,1,0;0;0;0;
60 FOR X=0 TO 1300 STEP 5
70 MOVE 0,300
80 DRAW X,300+300*SIN(X/103)
90 NEXT X
100 CLG:GOTO 40
```

PLASMATOID ORGANISM

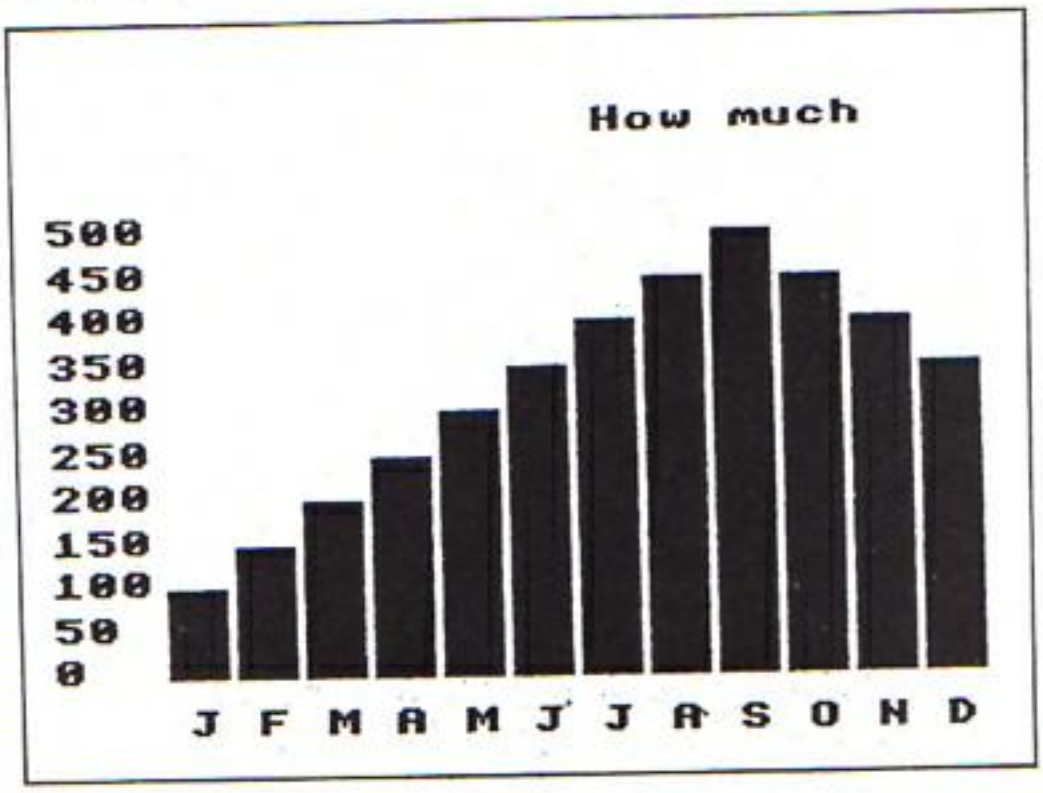
If you don't know what one is, Peter O'Brien's program will show you.

```
  10REM PLASMATOID
  20REM ORGANISM
  30REM PETER O'BRIEN
  40REM MOLD CLWYD
  50MODE2:COLOUR3
  60VDU23,1,0;0;0;0;
  70PRINTTAB(2,3)"***PLASMATI
OD***"
  80PRINTTAB(2,4)"****ORGANIS
M****"
  90GCOL0,6
 100MOVE0,645:DRAW1279,645
 110COLOUR2
 120SOUND1,-5,RND(255),2
 130VDU23,244,RND(255),RND(25
5),RND(255),RND(255),RND(255),
RND(255),RND(255),RND(255)
 140PRINTTAB(10,10);CHR$244
 150VDU23,224,36,36,36,231,23
1,0,0,0:COLOUR1
 160PRINTTAB(10,11);CHR$224
 170GOTO110
```

SIMPLE GRAPH

Ian Whitfield's elementary bar chart program.

```
10 REM SIMPLE GRAPH
20 REM IAN WHITFIELD
30 REM HUNTINGDON
40 MODE 1
50 VDU23,1,0;0;0;0;
60 PROCprint
70 GCOL 0,1
80 FOR X=115 TO 850 STEP 64
 90 INPUT TAB(25,2)"How much
",Y
 100 PRINTTAB(33,2)STRING$(5,
" ")
 110 PROCcol(X,Y)
 120 NEXT X
 130 END
 140 DEFPROCcol(X,Y)
 150 Y=Y+100
 160 MOVE X,100:MOVE X+50,100
 170 PLOT 85,X,Y:PLOT 85,X+50
,Y
 180 ENDPROC
 190 DEF PROCprint
 200 COLOUR 2
 210 PRINT TAB(4,31)"J F M A
M J J A S O N D"
 220 FOR S=120 TO 620 STEP 50
 230 VDU5:GCOL 0,2:MOVE 0,S
 240 READ N%: PRINT;N%:VDU4
 250 NEXT S
 260 ENDPROC
 270 DATA 0,50,100,150,200,25
0,300,350,400,450,500
```



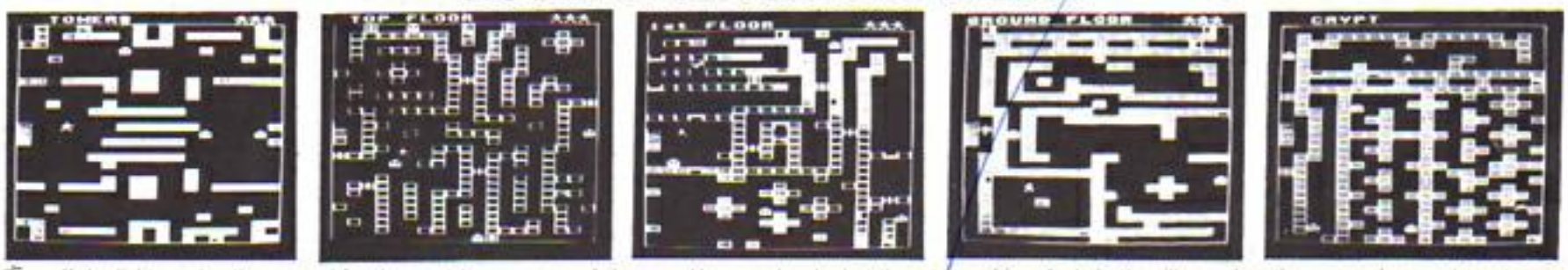
Send your programs to Scrapbook, *Electron User*, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

KAY-ESS COMPUTER PRODUCTS

PROFESSIONAL PROGRAMS FOR THE MODEL B AND ELECTRON

EACH TAPE ONLY £5.95 (Except H.O.H.)

HOUSE OF HORRORS (B)(E) £6.95.



Turn off the lights and gather around for the most creepy game of the year. How you laughed at those superstitious fools in the village when they warned you not to go near the old house. The climb up the rocky path under the afternoon sun was swift and within an hour you had passed through the outer gates of this once great house. The dust and cobwebs hadn't bothered you as you climbed the old stairs to the towers on the top level. Did you notice how low the sun had fallen before the sounds of locks clicking reached your startled ears? How can the moon be out already and what's that moving towards you??? This all action game will have you ducking and diving from the GHOSTS and ZOMBIES, and matching wits with a MUMMY, WEREWOLF, and VAMPIRE. 5 floors full of odd CORRIDORS, BROKEN FLOORBOARDS, and riddled with SECRET PASSAGES await you. Superb sound effects and graphics. Can be played using either keyboard or joysticks. Top table. Pause option.

EARLY YEARS (B)(E) For children between 3-6 years of age. These two packages give an adult or older child a means to take a younger child through a series of simple game type tasks to enforce ideas. The emphasis is on learning through fun. Topics covered include subtraction, addition, recognition, colour, shapes, sizes, sounds/notes, co-ordination, distances, estimates, directions.

EARLY YEARS 1
A) MICKEY THE MONKEY and his apple tree make subtraction fun.
B) COLOUR BLOCKS bring sizes and colour into perspective.
C) MERRY MUSIC turns the keyboard into a musical keyboard.
D) FUNNY FACES presents a line up, which one is the suspect?
E) FRED THE FROG needs co-ordinated help to get across the pond.

EARLY YEARS 2
A) THE POND seems very active today
B) SPEED is required to keep the cake on the conveyor belt.
C) DIRECTIONS seem to be needed by everyone in Orion village.
D) ORDER the blocks.
E) SID THE SPIDER needs some help to get out of the maze.

Watch out for HOUSE OF HORRORS at local dealers.

Dealer enquiries welcome.

All prices are FULLY inclusive for UK orders. Please add £1 per tape for non-UK addresses.

Cheques/P.O.'s should be made payable to

KAY-ESS Computer Products.

When ordering please state BBC or Electron.

Available for:
(E) Electron (B) BBC Model B
FREE with all orders (Except H.O.H.)
our 3 level version of
NOUGHTS AND CROSSES!!!

ALSO AVAILABLE:

STAR HAWKS (B) (E) - DESIGN (B) (E) - HANGMAN (B) (E) - SPACE TRAFFIC CONTROLLER (B) (E) - HORSES (B) (E) - SPACE TANK (B)

KAY-ESS Computer Products,
11 Buttercup Close,
Romleighs Park,
Harold Wood,
Essex RM3 0XF.

SIR COMPUTERS LTD CARDIFF

NEW

NOW AVAILABLE FOR THE electron

SEE US AT
The 2nd Official Acorn User Exhibition
Olympia 16-19 August 1984
Stand 104.

THE SIR COMPUTERS PRINTSTICK

ONLY £44.95

COMPLETE JOYSTICK & PRINTER FEATURES INCLUDE:

JOYSTICK FACILITIES
- Provides connections for two standard Atari-type joysticks, allowing the use of two-player games.
- Immediately compatible with all games offering a joystick option.
- Extra commands allow joysticks to be defined as any combination of keys, allowing all keyboard-operated games to be used with joysticks.
- Joysticks may be read directly from BASIC using the ADVAL(n) function.

PRINTER FACILITIES:
- Provides connections for a standard Centronics-type printer.
- Allows use of all BBC Microcomputer printer control commands.
- Special command enables a graphics screen to be copied to any Epson dot-matrix printer.

ADDITIONAL SPECIFICATIONS:
- Only Acorn-approved memory locations are used, ensuring complete compatibility with any future devices (sideways ROM/RAM, sound expansion, speech synthesiser, disc system etc.)
- All operating software is held within a paged ROM and is available for use from the moment the computer is switched on. THERE IS NO NEED TO LOAD ANY ADDITIONAL SOFTWARE FROM CASSETTE.
- Housed in a sturdy plastic case.
- Full twelve month guarantee.
- Available direct from SIR COMPUTERS for only **£44.95** (inc. VAT). Please include £1 postage and packing. Please allow 28 days for delivery.

SIR COMPUTERS – First for electron support

All our prices are inclusive of VAT. Please include £1 postage and packing for each item included on this page.
We also stock a complete range of printers, monitors, disc drives and software – with many hard-to-beat prices. Please telephone us for details.
Access/Barclaycard Telephone orders welcome.

SIR COMPUTERS LTD.
91 Whitchurch Road, Cardiff CF4 3JP. Telephone: Cardiff (0222) 621813

Also available for the Electron
THE SIR ADC/PRINTER INTERFACE

NOT JUST ANOTHER JOYSTICK PORT – FULL ANALOGUE-TO-DIGITAL CONVERTER provides fully proportional control, essential for use with graphics packages, digitizers, etc; ideal for scientific & educational applications; usable with a wide variety of BBC Micro-compatible analogue and switched Joysticks/Paddles.

CENTRONICS PRINTER INTERFACE – allows use of a wide variety of parallel printers including entire Epson range; complete firmware support included.

HIGH-QUALITY MOULDED CASE – attractively styled plastic unit bolts securely to the back of the computer.

EASY TO FIT – no soldering, simply plugs straight into computer's rear edge-connector and is held in place by twin bolts; edge-connector on back of unit provides for further modular expansion if necessary.

THE SIR ELECTRON PRINTER/ADC INTERFACE £65.95

THE SIR ROM/RAM EXPANSION BOARD

Provides 12 extra sockets which support a variety of ROM and RAM configurations up to a max of 192K for ROM and 16K for RAM.

ROM and RAM is normally paged in 16K blocks but is easily switchable to 2K, 4K or 8K blocks.

Easy to install – just plugs in.

Professional styled casing bolts to rear of computer.

Fully buffered design.

Permits use of most BBC ROM-based software including utility ROMs, wordprocessors & languages.

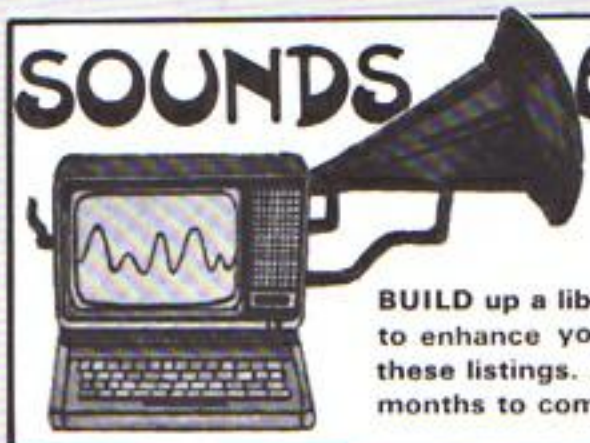
Price: £59.95

NOW COMBINED! Both above units (ROM/RAM Expansion Board and Printer/ADC Interface) in one case!

A complete and comprehensive Electron Expansion – ideal for word processing applications amongst many other uses.

ONLY £99!!

SOUNDS EXCITING



BUILD up a library of exciting sounds to enhance your own programs with these listings. And many more in the months to come!

Raindrops

From Nathan Perrin, Bradford

```
10 ENVELOPE 6,2,100,100,100,100
100,100,126,0,0,-126,126,126
20 SOUND 1,6,100,200
30 GOTO 20
```

The winner

From Graham Barlow, Stockport

```
10 ENVELOPE 1,2,-56,5,77,45,-99,
-2,126,0,0,-126,126,126
20 SOUND 3,1,157,180
30 GOTO 20
```

Rock music

From Paul Jones, Ipswich

```
10 ENVELOPE 6,128,123,155,135,
53,212,126,126,0,0,-126,126,126
20 SOUND 1,6,10,100
```

Invader crash

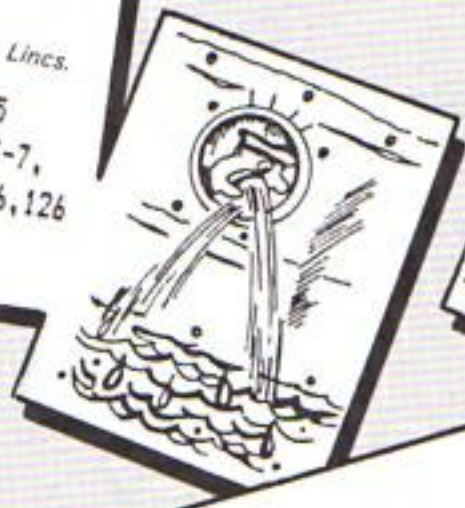
From M. Gallagher, Preston, Lancs.

```
10 REM INVADER CRASH
20 FOR A=150 TO 1 STEP -1
30 IF A=1 GOTO 70
40 ENVELOPE 4,1,10,25,10,5,
5,5,126,0,0,-126,126,126
50 SOUND 1,4,A,1
60 NEXT A
70 SOUND 0,-15,245,45
```

Seasick

From T. Blakely, Heckington, Lincs.

```
10 FOR X=35 TO 65 STEP 5
20 ENVELOPE 1,2,X,-X,X,X-7,
-X-7,X-7,126,0,0,-126,126,126
30 SOUND 1,1,X-4,40
40 NEXT X
```



Tinkling notes

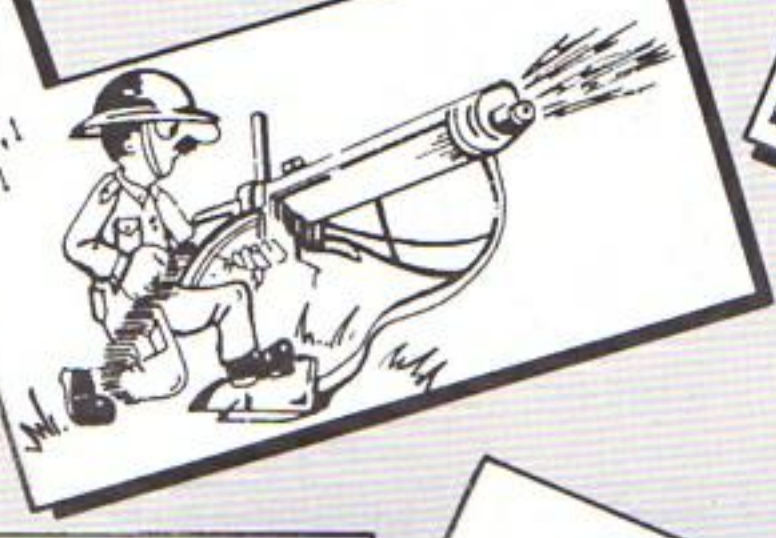
From Joelle Nicole Roccia, Carlisle

```
10 ENVELOPE 6,9,19,13,-8,4,
4,4,126,0,0,-126,126,126
20 SOUND 6,6,150,50
```

Machine gun

From M. Smith, Southend

```
10 REM MACHINE GUN
20 FOR N=1 TO 30
30 SOUND 0,-15,4,1
40 SOUND 0,0,4,1
50 NEXT
```



A ship sinking

From Stuart Monger, Billericay, Essex

```
10 SOUND 838,-17,RND(767),1
20 GOTO 10
```

Close encounters

From A.S. Florey, Derby

```
10 SOUND 1,-15,97,10
20 SOUND 1,-15,105,10
30 SOUND 1,-15,89,10
40 SOUND 1,-15,41,10
50 SOUND 1,-15,69,20
```

Do you have any sounds for Sounds Exciting? Send them into Electron User and hear yourself in print. The address: Sounds Exciting, Electron User, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

National Micro

Everything on this page is 5% less than our normal price

This special offer is exclusively for readers of Electron User and applies to mail order sales only.

We will also send you FREE membership of NMC's Computer Club – enabling you to enjoy generous discounts on all your future purchases!

Benefits for Club members include a big saving of 10% on software and 5% on hardware purchases over £25.

Personal shoppers are welcome at our retail stores:

National Micro Centres,
36 St. Petersgate,
Stockport SK7 5NY.
Tel: 061-429 8080

Wilmslow Micro Centre,
62 Grove Street,
Wilmslow, Cheshire.
Tel: 0625 530891

PRINTERS

Now you can add a printer to your Electron, using Plus 1, we have selected four of the most popular dot-matrix printers. All allow you to condense or embolden text, offer high definition characters and allow you to produce clear-cut graphics and charts:

Brother HR5 (30cps) **£170.95**
Epson RX80 (100cps) **£272.00**
Epson RX80FT (incl. friction feed) **£315.00**

For superb correspondence-quality printing you need a daisywheel printer. Our choice is one of our best-sellers, the Silver Reed EX43. It can also be used as a superior standalone electronic typewriter **£394.25**

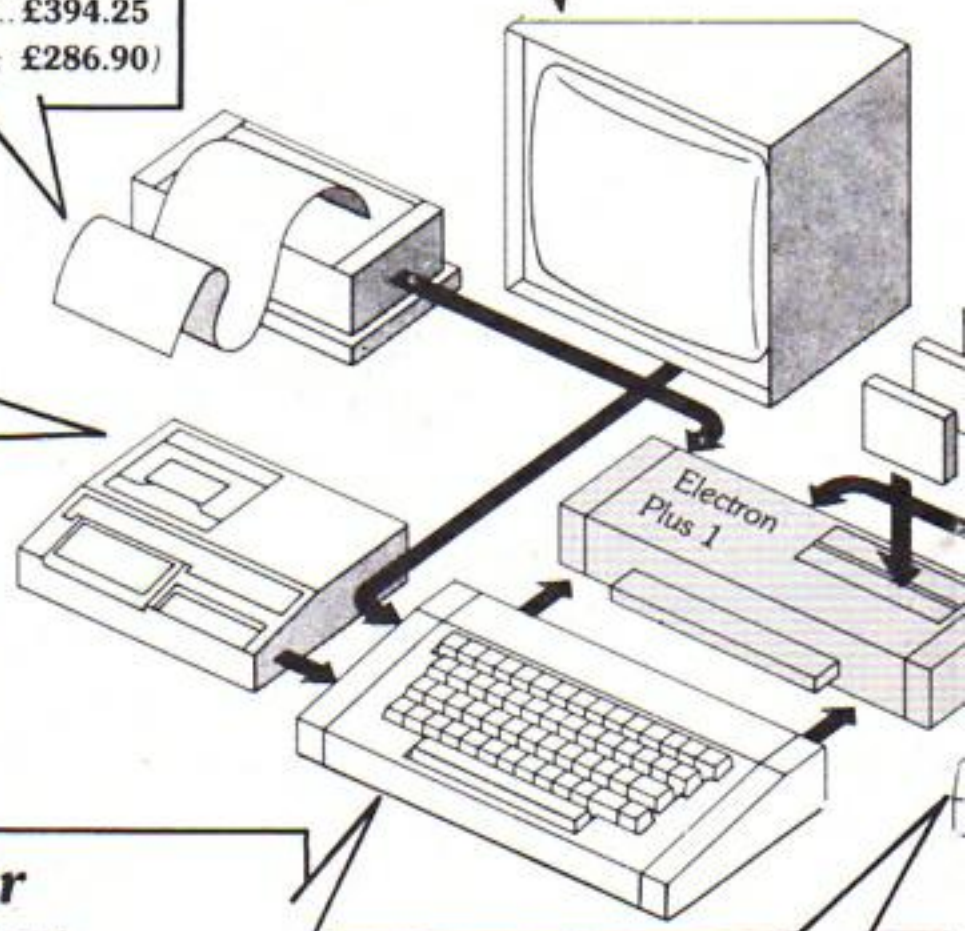
(Without Electron interface: **£286.90***)*

MONITORS

You can happily operate your Electron with your domestic TV set. But more and more users are finding that for a really crisp picture you need a special monitor. We offer a monochrome and three colour monitors:

Zenith 12" (green screen) **£81.00**
Microvitec (14" colour – low res) **£217.41**
Microvitec (14" colour – med res) **£326.66**
Microvitec (14" colour – hi res) **£480.70**

For the best of both worlds there is the 14" Nordmende, which can double as a monitor and normal TV, at a very attractive price **£238.00**
(with remote control **£251.00***)*



DATA RECORDER

From a wide selection of cassette recorders we recommend the Pye Data Cassette Recorder, which is a perfect match for the Electron. With it comes a FREE power pack and Electron lead.

£38.00

A FREE dust cover with every Electron

We have ample stocks of Electrons and can promise mainland delivery within 24 hours of receiving your order. With it comes an introductory cassette of 15 programs, a very comprehensive User Guide, an easy-to-understand DIY book on programming AND a free dust cover with the compliments of National Micro Centres **£189.00**

Electron Dust Cover *if supplied separately* ***£2.80***

JOYSTICKS

Use a joystick to play arcade games and watch your score increase dramatically! For serious games a joystick really is a must – and we have two we specially recommend. Both provide twin fire buttons.

Sureshot (self-centering action) .. **£15.67**
Clares (non self-centering) .. pair **£19.50**

These will take Kempston & Sureshot joysticks.

Selling well . . . First Byte's switched joystick interface

Since it was launched at the Electron & BBC Micro User Show the switched joystick interface from First Byte has been one of our top sellers. This plug-in cartridge takes standard Atari-style joysticks which are much more popular – and cheaper – than analogue joysticks. **£23.70**

Centres

MAIL ORDER DIVISION

Phone your order on **061-429 8080**

Answering service outside normal office hours
Or use the order form below

AT LAST! Plus 1 is the Electron add-on we've all been waiting for!

ELECTRON PLUS 1 is Acorn's answer to a growing demand from Electron users to be able to extend their micro's capabilities. With it you can add a printer and use your Electron for word processing and financial calculations. Its joystick input is designed to take two fully-proportioned joysticks – giving an entirely new dimension to games playing. And its two unique cartridge slots enable you to plug in games, educational and business programs – and that means no more waiting for programs to load. Many other manufacturers are now planning cartridges that will use Plus 1 to expand the Electron in many more exciting ways and considerably increase its power and versatility.

ELECTRON PLUS 1 is a must for every user who wants to really make the most of his micro.

Incredible value at **£56.90**

ROM CARTRIDGES

With Plus 1 you can use software cartridges on your Electron for the first time. Acornsoft has produced an initial range of cartridge games, educational and computer language programs, and many more will follow.

DELIVERY CHARGES
Hardware: £7 per item
Software: FREE

ALL PRICES GIVEN HERE INCLUDE VAT

Our Top Ten Best Sellers

Birds of Prey (Romik)
A fast moving invaders type game where the aliens in space take the form of birds. Great value for money. **£6.99**

Pharoah's Tomb (A & F)
Seek the golden mask in this graphic adventure, solve anagrams and number puzzles – but avoid the monsters. **£7.15**

Killer Gorilla (Micropower)
Fast becoming a cult game. Dodge tumbling barrels and blazing fireballs. Gripping multi-level action. **£7.95**

Twin Kingdom Valley (Bug-Byte)
A sophisticated adventure game with all 175 locations drawn in full-screen hi-res graphics. **£8.55**

Cylon Attack (A & F)
"Outstanding ... quite simply excellent ... the graphics leave most other games standing". – *Electron User* **£7.15**

Chess (Acornsoft)
One of the best computer versions of the game, easy to use, with more options than its competitors. **£8.28**

Felix in the Factory (Micro Power)
Never a dull moment for Felix, left in charge of the factory one evening. A great fun program. **£7.15**

Snapper (Acornsoft)
Gobble dots and fruit as you're chased round the maze by bog-eyed meanies. A real classic. **£8.28**

Starship Command (Acornsoft)
Guide your craft through deep space and avoid an enemy bent on your destruction. Very addictive. **£8.28**

Chuckie Egg (A & F)
A progressive game requiring extremely high skill levels. The nightmare has begun! **£7.90**

ORDER FORM

Post to:
NATIONAL MICRO CENTRES,
36 St. Petersgate,
Stockport SK1 1HL

Please supply the following:

Item	Qty	Total £	p
.....	.....	.....	
.....	.....	.....	
.....	.....	.....	
.....	.....	.....	
.....	.....	.....	
.....	.....		
		.....	

Attractive credit terms
Phone for details

Carriage .....
TOTAL .....

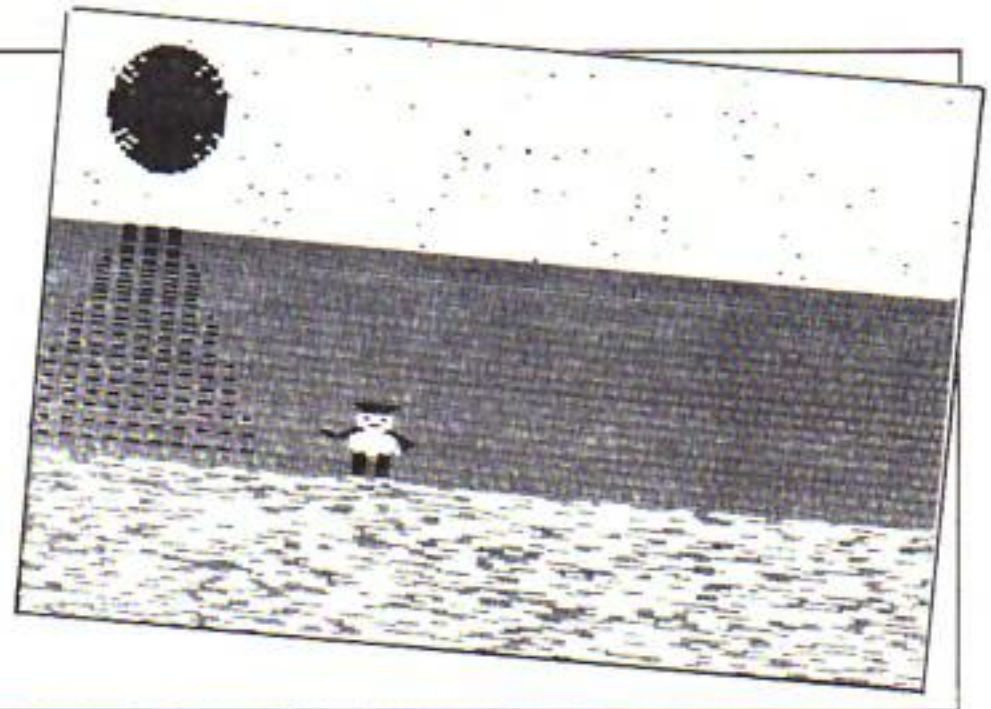
Please indicate method of payment:

☐ Cheque payable to National Micro Centres

☐ Access/Barclaycard No.

Name .....
Address .....
.....
.....
Tel. No. .....
Signed EU12

What shall we do with a dancing sailor?



SAILOR could almost be described as an all-singing, all-dancing program.

Actually it doesn't sing – just plays the hornpipe while an animated little sailor does nautical jigs.

Written by DAVID DAVIES of Glamorgan, it's a fine example of what can be achieved with Electron graphics.

```
  10 REM SAILOR
  20 REM BY DAVID DAVIES
  30 REM (C) ELECTRON USER
  40 MODE5
  50 VDU5
  60 VDU23,1,0;0;0;0;19,1,5,0
,0,0,19,2,4,0,0,0
  70 GCOL0,130:CLG
  80 X%=450:Y%=296:N%=0:newdr
awX%=0:dur=4
  90 VDU23,224,255,126,60,0,3
6,0,24,0
 100 VDU23,225,0,0,0,126,90,1
26,102,60
 110 VDU23,226,219,231,247,24
7,118,126,60,60
 120 VDU23,228,0,1,3,6,6,2,0,
0
 130 VDU23,229,0,0,0,0,0,1,1,
0
 140 VDU23,230,0,128,192,96,9
6,64,0,0
 150 VDU23,231,0,0,0,0,0,128,
128,0
 160 VDU23,232,102,102,102,10
2,102,102,102,0
 170 VDU23,233,0,0,0,0,0,0,0,
231
 180 VDU23,236,126,231,195,23
1,126,60,60,0
 190 VDU23,237,0,0,0,0,0,0,0,
126
 200 VDU23,238,0,17,27,14,0,0
,0,0
 210 VDU23,239,16,0,0,0,0,0,0
,0
 220 VDU23,240,0,136,216,112,
0,0,0,0
 230 VDU23,241,8,0,0,0,0,0,0,
0
 240 VDU23,242,126,230,198,23
0,118,54,54,0
 250 VDU23,244,126,103,99,103
,110,108,108,0
 260 VDU23,250,255,255,255,25
5,255,255,255,255
 270 VDU23,243,0,0,0,0,0,0,0,
119
 280 VDU23,245,0,0,0,0,0,0,0,
238
 290 VDU23,246,0,0,0,0,0,0,0,
0
 300 PROCscene:PROCmain:PROCa
rmone:PROCarmtwo:PROClegone:PR
OClegtwo
 310 REPEAT:PROCmain:PROCarmo
ne:PROCarmtwo:PROClegone:PROCl
egtwo:FOR N%=1 TO 4:PROCfirst:
PROCmove:PROCrepeat:PROCsecond
:PROCmove:RESTORE830:PROCrepea
t:PROCmove:dur=dur-1:RESTORE82
0:NEXT
 320 VDU25,4,X%+64;Y%;127,10,
9,9,127,127,127,10,9,9,9,127,1
27,127:X%=450:dur=4:TIME=INKEY
(200):UNTIL FALSE
 330 END
 340 DEFPROCfirst
 350 FOR T%=1 TO 25
 360 READ P%
 370 IF T%=3 OR T%=4 OR T%=5
OR T%=16 OR T%=17 OR T%=18 OR
T%=25 THEN D=dur*2 ELSE D=dur

 380 SOUND1,-15,P%,D
 390 SOUND1,0,P%,1
 400 PROCdraw
 410 NEXT T%
 420 ENDPROC
 430 DEFPROCrepeat
 440 FOR T2%= 1 TO 29
 450 IF T2%=27 OR T2%=28 OR T
2%=29 THEN D=dur*2 ELSE D=dur

 460 READ P%
 470 SOUND1,-15,P%,D
 480 SOUND1,0,P%,1
 490 PROCdraw
 500 NEXT T2%
 510 ENDPROC
 520 DEFPROCsecond
 530 FOR T3%=1 TO 26
 540 IF T3%=11 OR T3%=12 OR T
3%=13 OR T3%=24 OR T3%=25 OR T
3%=26 THEN D=dur*2 ELSE D=dur

 550 READ P%
 560 SOUND1,-15,P%,D
 570 SOUND1,0,P%,1
 580 PROCdraw
 590 NEXT T3%
 600 ENDPROC
 610 DEFPROCdraw
 620 drawX%=RND(8):IF drawX%=ne
wdrawX% GOTO620 ELSE ON drawX% G
OTO 630,640,650,660,670,680,69
0,700
 630 PROCarmone:GOTO710
 640 PROCarmtwo:GOTO710
 650 PROCarmthree:GOTO710
 660 PROCarmfour:GOTO710
 670 PROClegone:GOTO710
 680 PROClegtwo:GOTO710
 690 PROClegthree:GOTO710
 700 PROClegfour
 710 newdrawX%=drawX%:ENDPROC
 720 DEFPROCmove:VDU25,4,X%+6
4;Y%;127,10,9,9,127,127,127,10
,9,9,9,127,127,127:X%=X%+50:PR
OCmain:ENDPROC
 730 DEFPROCmain:VDU25,4,X%;Y
%;18,0,1,225,8,18,0,3,224,8,10
,226,8,18,0,0,227,8,8,18,0,3,2
28,8,18,0,1,229,9,231,8,18,0,3
,230,8,8,10,18,0,3,232,8,18,0,
1,233:ENDPROC
 740 DEFPROCarmone:VDU25,4,X%
;Y%-32;127,18,0,3,238,8,18,0,1
,239:ENDPROC
 750 DEFPROCarmtwo:VDU25,4,X%
+128;Y%-32;127,18,0,3,230,8,18
,0,1,231:ENDPROC
 760 DEFPROCarmthree:VDU25,4,
X%;Y%-32;127,18,0,3,238,8,18,0
,1,239:ENDPROC
 770 DEFPROCarmfour:VDU25,4,X
%+128;Y%-32;127,18,0,3,240,8,1
8,0,1,241:ENDPROC
 780 DEFPROClegone:VDU25,4,X%
+64;Y%-64;127,18,0,3,232,8,18,
0,1,233:ENDPROC
 790 DEFPROClegtwo:VDU25,4,X%
+64;Y%-64;127,18,0,3,236,8,18,
0,1,237:ENDPROC
 800 DEFPROClegthree:VDU25,4,
X%+64;Y%-64;127,18,0,3,242,8,1
8,0,1,243:ENDPROC
 810 DEFPROClegfour:VDU25,4,X
%+64;Y%-64;127,18,0,3,244,8,18
,0,1,245:ENDPROC
 820 DATA 120,112,120,72,72,1
00,92,88,100,120,116,120,136,1
28,120,128,80,80,80,72,68,80,1
00,96,100
 830 DATA 108,116,120,116,108
,100,108,100,92,88,92,88,80,72
,72,68,60,52,60,72,68,80,72,88
,80,92,88,72,72
 840 DATA 100,92,88,100,120,1
00,92,100,120,100,108,92,92,10
8,100,96,108,128,108,96,108,12
8,108,116,100,100
 850 DEFPROCscene
 860 GCOL0,0:PLOT4,0,1023:PLO
T4,1279,1023:PLOT85,0,500:PLOT
85,1279,500
 870 FOR N%=1 TO 300:A%=RND(1
279):B%=RND(523)+500:GCOL0,RND
(3):PLOT69,A%,B%:NEXT N%
 880 FOR S%=200 TO 0 STEP -4:
FOR T%=RND(20) TO 1279 STEP RN
D(20)+20:VDU18,0,RND(3)-1,25,4
,T%;S%;25,1,50;0;:NEXT T%:NEXT
 S%
 890 A=150:B=650:C=80:GCOL0,3
:MOVE A+C,B:FOR D=0 TO 2*PI+0.
05 STEP 0.05:MOVE A,B:PLOT5,A+
(C*COS(D)),B+(C*SIN(D)):NEXT D
 900 Y=500:REPEAT:VDU18,42,1,
25,4,180;Y;25,1,-(650-Y)/2;0;2
5,4,180;Y;25,1,(500-Y)/2;0;:Y=
Y-((550-Y)/15):UNTIL Y<=204
 910 ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

Make light work of listings

To save your fingers most of the listings in *Electron User* have been put on tape. Eight are now available – for the February, March, April, May, June, July, August and September issues, plus a bumper tape of all the programs from the introductory issues.

On the September tape:

HAUNTED HOUSE Arcade action in the spirit world. **SPLASH** A logic game for non-swimmers. **SORT SHOWS** How sorting algorithms work. **SORT TIME** The time they take. **CLASSROOM INVADERS** Multicoloured characters go to school. **SAILOR** Nautical antics. **MATHS TEST** Try out your mental powers. **MOVER** Keep that alien under control. **NOTEBOOK** Sound and graphics action.

On the August tape:

SANDCASTLE The Electron seaside outing. **KNOCKOUT** Bouncing balls batter brick walls. **PARACHUTE** Keep the skydivers dry. **LETTERS** Large letters for your screen. **SUPER-SPELL** Test your spelling. **ON YOUR BIKE** Pedal power comes to your Electron. **SCROLLER** Sliced strings slide sideways. **FLYING PIGS** Bacon on the wing. **FAST ELLIPSE** Speedy graphics. **NOTEBOOK** Lines and patterns explained.

On the July tape:

GOLF A day on the links with your Electron. **SOLITAIRE** The classic solo logic game. **TALL LETTERS** Large characters made simple. **BANK ACCOUNT** Keep track of your money. **CHARTIST** 3D graphs. **FORMULAE** Areas, volumes and angles. **NOTEBOOK** Time table.

On the June tape:

MONEY MAZE Avoid the ghosts to get the cash. **CODE BREAKER** A mastermind is needed to crack the code. **ALIEN** See little green men – the Electron way! **SETUP** Colour commands without tears. **CRYSTALS** Beautiful graphics. **LASER SHOOT OUT** An intergalactic shooting gallery. **SMILER** Have a nice day!

On the May tape:

RALLY DRIVER High speed car control. **SPACE PODS** More aliens to annihilate. **CODER** Secret messages made simple. **FRUIT MACHINE** Spin the wheels to win. **CHASER** Avoid your opponent to survive. **TIC-TAC-TOE** Electron noughts and crosses. **ELECTRON DRAUGHTSMAN** Create and save Electron masterpieces. **SHEEP** A program for insomniacs. **MATHS HIKE** Mental arithmetic. **MESSAGE** VDU commands in action.

On the April tape:

SPACEHIKE A hopping arcade classic. **FRIEZE** Electron wallpaper. **PELICAN** Cross roads safely. **CHESSTIMER** Clock your moves. **ASTEROID** Space is a minefield. **LIMERICK** Automatic rhymes. **ROMAN** Numbers in the ancient way. **BUNNYBLITZ** The Easter program. **DOGDUCK** The classic logic game.

On the March tape:

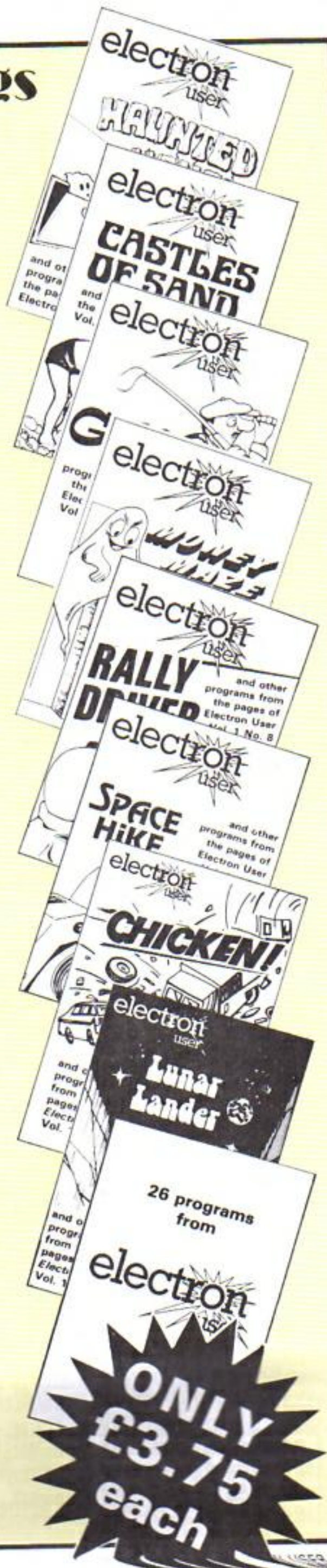
CHICKEN Let dangerous drivers test your nerve. **COFFEE** A tantalising word game from Down Under. **PARKY'S PERIL** Parky's lost in an invisible maze. **REACTION TIMER** How fast are you ? **BRAINTEASER** A puzzling program. **COUNTER** Mental arithmetic can be fun! **PAPER, SCISSORS, STONE** Out-guess your Electron. **CHARACTER GENERATOR** Create shapes with this utility. **FUNNY POLYGONS** Fast graphics going round in circles.

On the February tape:

NUMBER BALANCE Test your powers of mental arithmetic. **CALCULATOR** Make your Electron a calculator. **DOILIES** Multi-coloured patterns galore. **TOWERS OF HANOI** The age old puzzle. **LUNAR LANDER** Test your skill as an astronaut. **POSITRON INVADERS** A version of the old arcade favourite. **MOON RESCUE** Avoid the asteroids and save the spacemen.

On the introductory tape:

ANAGRAM Sort out the jumbled letters. **DOODLE** Multicoloured graphics. **EUROMAP** Test your geography. **KALEIDOSCOPE** Electron graphics run riot. **CAPITALS** New upper case letters. **ROCKET, WHEEL, CANDLE** Three fireworks programs. **BOMBER** Drop the bombs before you crash. **DUCK** Simple animation. **METEORS** Collisions in space. **COMBINATIONS** Crack the code. **BUZZ WORD GENERATOR** Let the Electron help you impress.



HOW TO ORDER

Please send me the following *Electron User* cassette tapes:

Nine programs from the September issue ..... £
Fourteen programs from the August issue ..... £
Ten programs from the July issue ..... £
Ten programs from the June issue ..... £
Twelve programs from the May issue ..... £
Eleven programs from the April issue ..... £
Twelve programs from the March issue ..... £
Nine programs from the February issue ..... £
26 programs from the introductory issues ..... £ _____

I enclose the sum of £ _____

Name .....

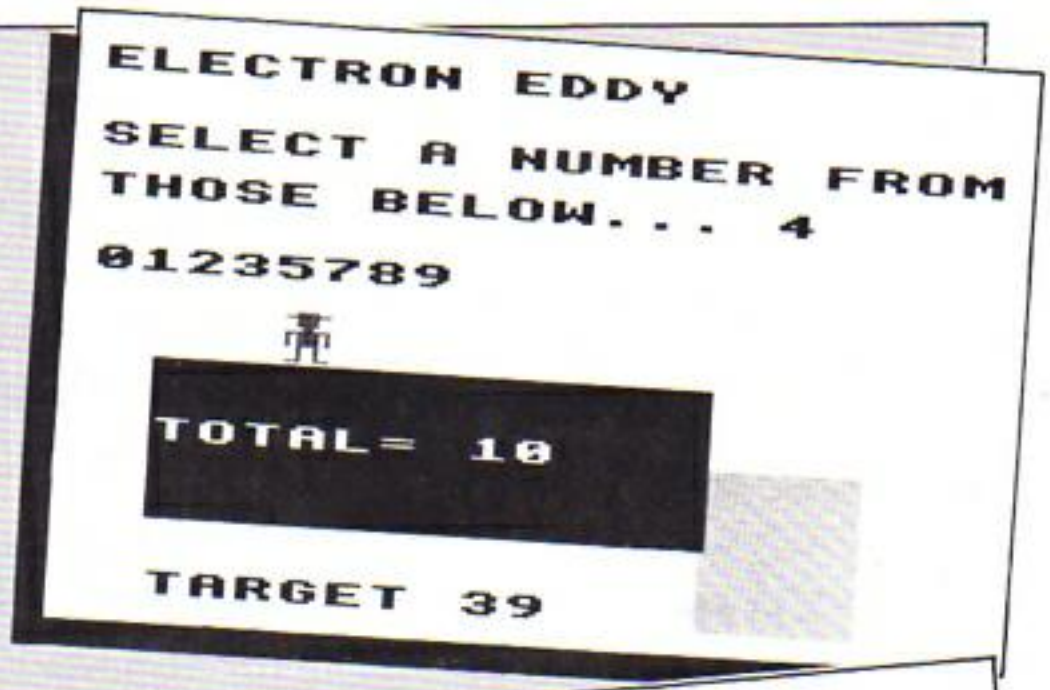
Address .....

.....

POST TO: Tape Offer, Electron User, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Take the right steps or you'll make a . . .

SPLASH!



SPLASH, by ROGER FROST, is a two-player game of mathematical strategy that will entertain and intrigue both adults and children.

The idea is that a little man moves across a promenade towards the sea. He can only take a certain number of steps – chosen by the Electron – before he falls into the water.

Each time it is your turn, you must choose how many steps he takes by selecting a number from nought to nine.

Players take turns to pick a number and once that number has been chosen it can't be used again. Meanwhile, the little man gets nearer and nearer the edge.

The idea is to force your opponent to pick a number that will make him fall into the sea.

As you can imagine, it's all good, clean fun. The target figure, running total and the numbers available are all displayed on screen, allowing the players to concentrate on strategy.

The controls are simple. You just type in the number that you want and press Return. The rest is up to you. Don't get wet!

VARIABLES

PROCinit	Sets up the variables, text and graphics windows, and draws the starting screen.
PROCtext	Sorts out whose go it is and receives a valid input number.
PROCgraphics	Updates the position of the man and keeps a record of the running total.
PROCsplash	Makes the man fall in the water if the target number is exceeded. It informs the loser and invites you to play again.

```
10REM***SPLASH***
20REM***BY ROG FROST***
30REM (C) ELECTRON USER
40MODE6
50PROCrules
60MODE2
70FORgame%=1TO10
80PROCinit
90PROCtext
100NEXT
110MODE6
120PRINT'''player$(1);" won
";score%(1);" games."
130PRINT'''player$(2);" won
";score%(2);" games."
140END
150DEFPROCinit
160C=0
170pos=-20
180prom=30+RND(14)
190VDU23;8202;0;0;0;
200VDU28,0,17,19,0
210VDU24,0;0;1279;444;
220VDU19,1,4,0,0,0
230GCOL0,134:CLG
240VDU29,100;100;
250VDU23,228,60,255,60,60,60
,24,255,189
260VDU23,229,189,189,189,36,
36,36,36,231
270MAN$=CHR$228+CHR$10+CHR$8
+CHR$229
280GCOL0,3:MOVE0,0:MOVEprom*
20,0:PLOT85,0,200:PLOT85,prom*
20,200
290GCOL0,4:MOVEprom*20,100:M
OVE1000,100:PLOT85,prom*20,-10
0:PLOT85,1000,-100
300GCOL0,0:VDU5:MOVE-20,270:
PRINT;MAN$
310GCOL0,0:MOVE0,-70:PRINT"T
ARGET ";prom
320GCOL0,0:MOVE0,125:PRINT;"
TOTAL= ";C
330VDU4
340A$="0123456789"
350ENDPROC
360DEFPROCtext
370FORGO=A%TOA%+9
380PERSON=GO MOD2+1
390COLOUR135:CLS
400COLOUR 4
410PRINTTAB(0,1);player$(PER
SON)
420PRINTTAB(0,15);A$
430INPUTTAB(0,4)"SELECT A NU
MBER FROM"''"THOSE BELOW... "B
440IF B<0 OR B>9 PRINT''"Don
't be silly":J=INKEY(300):GOTO
390
450B$=STR$(B)
460FORN=1TOLEN(A$)
470C$=MID$(A$,N,1)
480IF C$=B$ N=LEN(A$)
490NEXT
500IFC$<>B$ PRINT''"Number n
ot there":J=INKEY(300):GOTO390
510X=INSTR(A$,C$)
520A$=LEFT$(A$,X-1)+RIGHT$(A
$,(LEN(A$)-X))
530PROCgraphics
540IFC>prom GO=A%+9
550NEXT
560ENDPROC
570DEFPROCrules
580DIMscore%(2),player$(2)
590A%=0:game%=0
600SOUND0,-15,20,50
610VDU19,0,4,0,0,0
620PRINTTAB(16,1)"SPLASH"
630VDU23,1,0;0;0;0;
640PRINT''"This is a two pla
yer game."''"Each player will t
ake a turn to pick a  number.
They will lose if the total of
 all numbers picked is bigger
 than the"''"target."
650PRINT"A running total is
kept so that"''"players will kn
ow where they are."''"Remember
to press RETURN to enter"''"a n
umber."
660PRINTTAB(0,14)"DON'T LET
THE MAN FALL INTO THE WATER."
670INPUT''"Enter first playe
r's name",player$(1)
680INPUT''"Enter second play
er's name",player$(2)
690ENDPROC
700DEFPROCgraphics
710VDU5
720GCOL0,6:MOVEpos,270:PRINT
;MAN$
730C=C+VAL(C$)
740pos=C*20-20
750GCOL0,0:MOVEpos,270:PRINT
;MAN$
760GCOL0,3:MOVE0,0:MOVEprom*
20,0:PLOT85,0,200:PLOT85,prom*
20,200
770GCOL0,0:MOVE0,125:PRINT;"
TOTAL= ";C
780IF C>prom PROCsplash
790VDU4
800ENDPROC
810DEFPROCsplash
820FORypos=270 TO150 STEP-20
830GCOL0,6:MOVEpos,ypos:PRIN
T;MAN$
840GCOL0,0:MOVEpos,ypos-20:P
RINT;MAN$
850NEXT
860GCOL2,4
870FORline=0TO20:MOVEpos+20,
ypos-50:DRAWpos+300-RND(600),y
pos+RND(200)
880SOUND0,-15,20,2
890NEXT
900VDU4
910CLS
920PRINTTAB(0,2)"SORRY! ";pl
ayer$(PERSON);" LOST"
930score%(PERSON)=score%(PER
SON)+1
940A%=A%+1
950IF game%=10ENDPROC:GOTO11
0
960PRINTTAB(0,12)"Press SPAC
E to play"
970REPEATUNTILGET=32
980ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

Software Surgery

THE COLUMN THAT TAKES A LOOK INSIDE THE LATEST RELEASES

Plunge into the Abyss for a quiz, not adventure

Abyss
Cases Computer Simulations

ABYSS is described as an unusual adventure game, requiring you to undertake a succession of mental tests, aptitude tests and arcade games.

To achieve your goal, which is to get to the end of the trail, you must complete every test correctly. The tests are randomised as to type and difficulty so you can be lucky and have a succession of the easier tests.

However to achieve your goal you will have to prove your intellectual prowess and be able to get the more difficult tests right as well.

After each test you return to the matrix and proceed to the next stage. To reach the following step you have to cross over rickety bridges. If you're unlucky the bridge will fail and you will be plunged into the abyss.

It is a game for one player, the rules are simple and no problems were experienced in loading.

The front of the cassette is labelled CCS adventure games, but to my mind this is not really an adventure game.

As an adventure freak I was bound to dislike this program, since I had obviously expected an adventure. However having said that I could find no technical fault with it. It's a capable program, but not the adventure it purports to be, being more of a quiz than anything else.

Merlin

Hide, seek and learn

Invisible Man
Chalksoft

THIS is a sort of educational Battleships for eight to 13-year-olds. When you run the game a 10 by 15 grid is displayed and then disappears.

Somewhere in the 150 squares on the screen an invisible man is hiding. The aim is to expose him to view by guessing which squares he is hiding under.

The kids enter this into the Electron using a simple co-ordinate system. If they miss with their shot the micro responds with a hint, using the points of the compass.

This makes it much less of a game of chance than Battleships and allows the children to use and expand their knowledge of simple co-ordinates and compass directions.

The idea is to find the man in fewer tries than your rivals.

The whole thing is well explained, simple to use and extremely idiot-proof. It's also a pleasant game in its own right.

The only quibble is that it might have been better if the lines of the grid were left on the screen rather than disappearing when the man hides. This, I am sure, would enhance an already useful program.

Peter Gray

Keeping happy

Happy Numbers
Bourne Educational Software

THIS is another in the Happy Series by Bourne, and it maintains the promise of the first title in the series, Happy Letters.

It is aimed at a very young audience, this time between three and six, probably with an adult to help where necessary.

Again the sound can be turned off, and I tended to do that rather often.

A large number is drawn filling most of the left side of the screen. The child then enters an input by pressing the appropriate number key the correct number of times.

As each key press is made, a flower is added to the right hand side. If the correct input is made, the face smiles, and a flower is added to the score.

Should the answer be incorrect however, the flowers change colour and only a stalk is added to the score. A useful extra feature is that on an incorrect input the same large scale number is drawn so that the child might compare the two.

There is the usual monitor screen, which allows the adult to assess the progress of up to five children. Should particular problems appear, the sequence of numbers given to a child may be set in advance by the adult to strengthen weaknesses.

It is good to see that the Break key is at least partly protected, returning the user to the main menu. It is also encouraging that such good quality programs are being made available for the Electron, emphasising the fact that it is an ideal machine for education both at school and at home.

This program is good value and will prove useful for any young family.

Phil Tayler

From Page 27

Back with Spock

SpaceTrek
Dimax Structured Software

THIS is quite a good implementation of the classic game for micros. I would imagine there are still a few people who have not yet commanded the Starship Enterprise on its voyages through the edges of space where it encounters the dreaded Klingons/Vaiders.

The program offers a variety of commands via which the on-board computers produce information on screen. The commands are printed on handy reference cards included in the cassette case.

The program is written to run on both the Electron and the BBC Micro, so the speed tends to suffer when run on the Electron.

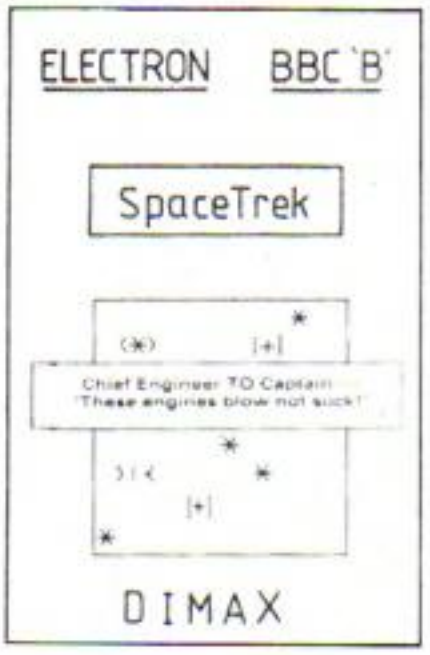
With the program priced competitively and the listing freely available, Dimax have bravely taken a positive stand in the controversy over copying.

I have played Trek on the Electron before, and this is a much better version than one I bought previously. Even with all the on-board computers, it's still very tough going.

There are moments requiring split-second decisions, but generally tactics can be formulated carefully.

The speed with which I moved from Condition Green to being disabled was rather disconcerting, but I think that was more a reflection on the captain than the ship.

Phil Tayler



Just one more go...

Blagger
Alligata

FEEL in the mood for some thieving? Like to rob a safe or two? Well Blagger gives you the chance.

As Roger the Dodger, intrepid master burglar, you have four lives and 20 different screens to search for golden keys in houses, shops and banks.

Spectrum owners have made a lot of fuss recently about a game called Jet Set Willy, but I'm certain that Blagger would give it a run for its money.

It's nothing if not addictive, and a great deal of patience and thought are required in order to work out how to get through each scene.

Some surfaces disappear as you walk over them, mysterious gooseberry-like objects kill you if you touch them and sundry nasties such as spaceships, humbugs, gnashing teeth, locomotives and RG signs must be avoided. Don't ask me what RG stands for, I don't know.

After collecting all the golden keys from the safes on one screen you have to reach a safety zone before the game progresses to the next level.

This is difficult if you've eroded away the only escape route while collecting the keys! Also there's a time limit for each screen, which adds to the fun.

The game is a winner and is very much a "just one more go" piece of software. The program makes good use of colour and graphics although the sound can sometimes get a little irritating.

The keys are easy to use, the instructions clear and the game itself very enjoyable, sure to give many hours of entertainment.

Steve Yarwood

Slap for sluggards

Adventure
Program Power

THE Electron acts as your eyes and ears in this all-text adventure game where you wander a fantasy world trying to rescue a princess from magic caverns.

You have a compass, but it is fairly unreliable, often making it almost impossible to retrace your steps. You're well advised to make a map as you go along.

The keyboard entry uses the standard verb/noun system such as "Take axe" or "Go West". The vocabulary available is fairly large though the game leaves you to discover this for yourself, only telling you the most basic commands.

One annoying feature is that the Electron can get bored

if you take your time – and it tells you so! This is usually when you're completely lost and it doesn't help. The game can be listed, but this provides no answers, which annoys me as I still can't get past the

Maths is an adventure

Adventure Quiz
Dial Soft

THIS is not, as one might think, a test of knowledge of adventures but an educational program in an adventure setting. Its purpose is to test children's knowledge of mathematics.

The program comes in two parts, the quiz itself and a drawing program which is automatically CHAINed if the questions set in the quiz are answered correctly.

The quiz makes little use of the graphics capabilities of the Electron and too much of the sound. I eventually used *FX210,1 to turn the sound off, though the kids loved it.

The program is well error-trapped and listable on loading. Being written in Basic, it is easy to adapt.

Since you are not given the correct result when a wrong answer is given, this would probably be the first thing you would change.

The questions cover multiplication up to the 12 times table, division, subtraction, addition and mixtures of these. Only whole numbers are used.

The drawing game offered as a reward for successfully answering the quiz is a simple etch-a-sketch type program. It isn't as well error-trapped as the quiz but since this is also in Basic that, too, can be easily changed.

The quiz won't teach children mathematics but it is novel enough to hold their attention and I found there was fierce competition to see who could get to the end first.

The reward stands up well against commercial drawing packages, though a separate instruction sheet would have been helpful.

A useful package that could be slightly improved, but is nevertheless good value for money.

Peter Lundstrom

fierce rat and you can't kill him yourself.

As the cassette insert tells you, your recollection of stories from the Arabian Nights should stand you in good stead. As you might expect, there are various treasures and objects in the adventure, including a bottle of potent wine. You can drink the wine but hang on to the bottle.

All in all, the game is well written and the answers may take some time to work out, but the clues are all there. It's the type of program that has people looking over your shoulder giving "helpful" hints and ideas to try.

Although perhaps a little simple for the experienced adventurer, for the beginner it is an extremely good game.

Neil Graham

Tactics to stay alive

Pengwyn
Postern

WHEN I first tried Pengwyn I wondered why the game was so ponderous. Then I spotted that the cassette is printed in identical fashion on both sides, but each side is dedicated to a different computer!

While one side is indeed for the Electron I had mistakenly loaded the BBC version. Still, the slower speed enabled me to develop some tactics so that when I loaded the correct side I managed to stay alive for a while.

Many of the best games are simple in concept, and one of my all-time favourites has been Jet Pac, which I enjoyed when I had my S*e*t*u*.

In Pengwyn the storyline is equally simple. The feathered friend stands surrounded by blocks of ice and three vibrantly shining eggs.

By melting and/or sliding the blocks, the Pengwyn has to get the three eggs in a straight line anywhere on the screen.

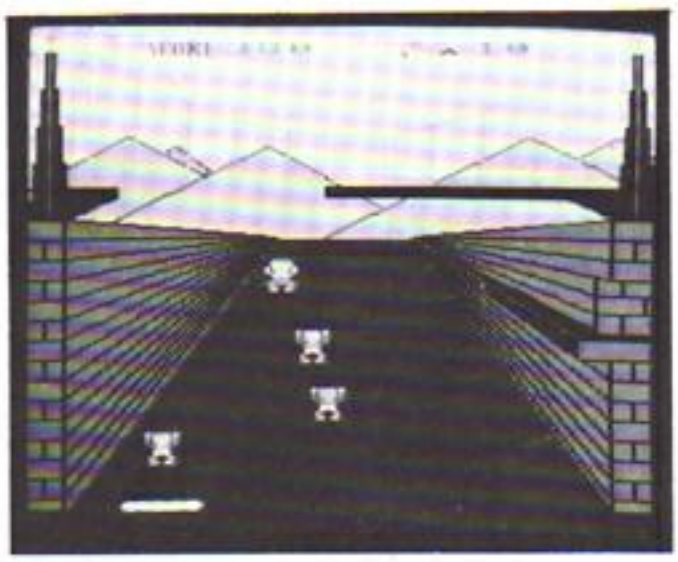
Sounds simple, doesn't it? The trouble is that a couple of yellow monsters melt out of the blocks and pursue the little bird, although their movements are predictable.

It's a lovely feeling when your Pengwyn slides a block which then crushes a monster flat – although another is always waiting to melt out.

What makes the program good value? I can't say it's the sound, which is barely adequate. But the animation is superb, with some delightful touches – like the bird's feet dancing around on the cold ice.

When the poor creature is trapped its look of total bewilderment and dejection is a masterpiece of comic pathos. With a high score table to keep tabs on your progress, it's the kind of program you will go back to again and again . . . and again.

Phil Tayler



Save a politician!

Lemming Syndrone
Dynabite Software

A GAME for those with fast reactions and a burning desire to save people from drowning, Lemming Syndrome is one of those compulsive games that always has you wanting one more go.

The idea is simple. Mad Marco, the world famous arsonist, has set light to a city and trapped the population between the flames and a deep, dangerous river. As is the way in computer games, none of the people can swim (remember the drowning frog in Croaker?).

Having said that, they would rather chance their arm in the water than in the flames, so, singly or in groups, they hurl themselves over the edge. This is where you come in, you and your little rubber raft.

If you place your raft beneath the plummeting population they bounce. The trouble is that they only bounce as far as the centre of the river. The result is that you have to catch them again and bounce them twice more before they reach dry land at the other side of the river. And you've got to watch out because while you're waiting for the third bounce you notice another group hurling themselves off the edge. Can you get back in time?

To make matters worse, Mad Marco keeps throwing sticks of dynamite (which you avoid) and there's a hungry shark (which you try to avoid).

The game ends when you've been blasted, bitten or lost 50 people. You get points for each person you save, the various classes having different scores. For some obscure reason politicians are worth more than anybody!

The graphics are excellent, the colours well thought out and the controls simple to use. You have a choice of sound on or off, various levels of skill and different ways of moving your rubber raft.

Calling for quick reactions and a sense of humour it's a game that will have you laughing as you press for another go. Great fun.

Trevor Roberts

Nature in the raw..

Savage Pond
Starcade

ARE you an ecology buff? Does your soul resonate to Mother Nature in all her glories? If so then Starcade is the game for you. Even if you're not a nature freak you'll probably still go for it.

By virtue of the ?, *, Z, and X keys you become a tiny tadpole, swimming round in a pond, eating amoebae to keep up your energy. An idyllic sounding life, isn't it?

The trouble is that it's not just you eating amoebae, it's other things eating you. The hydra that lurks on the bottom of the pond is just one example.

You can get temporary immunity to the hydra by gobbling down some of the little blue worms that are slowly falling through the water.

Every five of these that you gobble adds to your score and takes you one step nearer being a frog.

The aim of the game is to build up a colony of these frogs. It seems at times that everything else has the opposite intention.

Don't pay too much attention to the impressive-looking dragonfly buzzing overhead, it won't do you any harm. Having said that, you have to watch out for her eggs which if they get to the bottom, develop into a nasty monster with an enormous appetite.

And if that's not enough, as the game progresses there's also radioactive dumping, mutant bumble bees, cowardly water spiders, water fleas and a whole host of other dangers to your colony.

It's not easy being a tadpole in the savage pond but it is fun. The game is original, amusing and addictive. In fact it's a winner.

Bev Friend

Electronic News

Programming the Electron

A must for Acorn Electron users – a book to teach you how to make the most of the sophisticated features of this micro-computer.

After a short introduction to the machine and how to get it started, some general points on programming techniques are presented followed by more specific features of Electron BASIC including graphics facilities, string handling, mathematical functions, random numbers and sound. Subsequent chapters introduce bits and bytes, hexadecimal numbers and assembly language programming, interfacing features and file handling. Appendices cover technical specification, error messages, ASCII codes and the 6502 instruction set.

Softcover *176 pages* *£6.95 approx.*

Order now from your Bookseller or direct from

Newnes Technical Books
Borough Green, Sevenoaks, Kent TN15 8PH

Try to beat the clock in this mental arithmetic program by ALAN McLACHLAN

Time for a maths test

```
The number you enter will set
the length of time that
each sum stays on the screen.
E.G.
5 gives 10secs  for 1 point
1 gives  2sec   for 5 points
The faster you play.........
the more points you can score

IF YOU DO NOT ANSWER IN TIME
YOU SCORE AS A WRONG ANSWER.
```

IT seems the Electron is never satisfied. After all, it is the one that's supposed to have the mathematical brain – but in Maths Test, it asks YOU all the questions.

Still, it is fun trying to guess – I mean calculate – the answer before you run out of time.

And you can make the test as easy as you like.

PROCEDURES

PROCinst Instructions
PROCspeed Sets time question on screen
PROCnumbers Sets highest number in your range
PROCmenu Menu of mathematical operations
PROCadd Addition
PROCsub Subtraction
PROCtimes Multiplication
PROCscore Prints scoreboard
PROCans Tests right or wrong
PROCdelay (secs%) Variable delay
PROCsoundright Sound for right answer
PROCsoundwrong Sound for wrong answer
PROCsoundnoans Sounds for no answer (out of time)
FNget_number (DL%,TL%) Checks that the only inputs are numbers and waits for (DL%,TL%) secs

MAJOR VARIABLES

A,B: Random number for maths
C: Result of maths
D: Your answer
ENDTIME: Time up
HI: Highscore
DF: Delay factor to make 1 sec.
SF: Length of time factor is on screen
R: Right answer
W: Wrong answer
T%: Time variable
time%: A flag-true if out of time
SPEED: Speed rate of input
NUMBER: Maximum range of input
BONUS: Bonus based on NUMBER
final$: Number input in string form
TIMENOW: Temporary storage for actual time while setting TIME to 0.

Maths Test listing

This listing was produced using a special formatter which breaks one program line over several lines of listing. When entering a line don't press Return until you come to the next line number. Full details of the formatter are given on Page 4 of the February issue.

```
10 REM MATHS
20 REM BY ALAN McLachlan
30 REM (C) ELECTRON USER
40 MODE 2
   :COLOUR 131
   :COLOUR 0
50 HI=0
60 CLS
   :VDU 23;8202;0;0;0;

70 PRINT TAB(3,14)"INSTRUCTI
   ONS?"
80 PRINT TAB(7,16)"Y / N
   "
90 *FX15,1
100 REPEAT
110 A$=GET$
120 UNTIL A$="Y" OR A$="N"
130 IF A$="N"
   THEN 200
140 MODE 1
150 VDU 23;8202;0;0;0;
160 PROCinst
170 *FX15,1
180 A$=GET$
190 MODE 2
   :COLOUR 131
   :COLOUR 0
200 SC=0
   :SCORE=0
210 REM **********************
   *****
220 REM **********************
   *****
230 REM **********************
   *****
240 SF=200
   : REM SPEED FACTOR
250 REM FOR BBC
260 DF=100
   : REM DELAY FACTOR
270 REM FOR BBC
280 MF=6000
   : REM MINUTE FACTOR

290 REM FOR BBC
300 REM **********************
   *****
310 REM **********************
   *****
320 REM **********************
   *****
330 A=0
   :B=0
   :C=0
   :Z=0
   :T=0
   :R=0
   :W=0
340 dummy=RND(-TIME )
   :REM seed random no.
    generator
350 CLS
   :VDU 23;8202;0;0;0;

360 PROCspeed
370 PROCnumbers
380 PROCmenu
390 *FX15,1
400 REPEAT
410 PRINT TAB(4,24)"YOUR
     CHOICE ";
420 Z=GET -48
430 IF Z<1 OR Z>3 PRINT
   TAB(1,24)"COME ON...
    1,2,OR 3";
   :PRINT TAB(1,25)STRING$(2
   0," ")
   :PROCdelay(2)
   :PRINT TAB(1,24)STRING$(2
   0," ")
440 UNTIL Z>0 AND Z<4
450 ON Z GOTO 460 ,470
   ,480
460 PROCadd
   : PROCscore
   : GOTO 190
470 PROCsub
   : PROCscore
   : GOTO 190
480 PROCtimes
   :PROCscore
   : GOTO 190
490 REM **********************
   *****
500 REM ADDITION
510 REM **********************
   *****
520 DEF PROCadd
530 *FX15,1
540 CLS
   :COLOUR 3
   :PRINT TAB(7,15)"READY?"
   :PROCdelay(2)
550 T%=TIME
   :ENDTIME=T%+MF
560 REPEAT
570 A=RND(NUMBER)
   :B=RND(NUMBER)
580 C=A+B
590 CLS
600 COLOUR 128
   :COLOUR 1
610 PRINT TAB(2,10)"WHAT
     IS ";A;" + ";B
620 PRINT '''''
```

Turn to Page 57

TRAPPED IN A HAUNTED HOUSE

HAUNTED House is a fast and challenging arcade style game for the Electron and BBC Model B written by PETER SCOTT.

If your nerves and reactions are up to it, you take the part of a man trapped in a haunted room being constantly harassed by spooks, spectres and – don't ask why – aliens.

To get out of the room the man must collect all the keys lying scattered round the screen. At the same time he has to get rid of all the nasties by shooting them with a laser.

He can then escape by running to the door at the top left of the screen. On the way he can gather various objects left around the room – and collect bonus points – but avoid the ghosts and cans.

You have a slight disadvantage in that the little man can only fire if you are facing sideways and, no matter what you do, the man keeps on moving.

The reward for all this is that you move onto a harder room with more obstacles, more treasure and up to 12 aliens.

The game features a high score table, sound on and off functions, a pause facility and, if you reach the fifth room, a bonus life so you can face more of the dead.

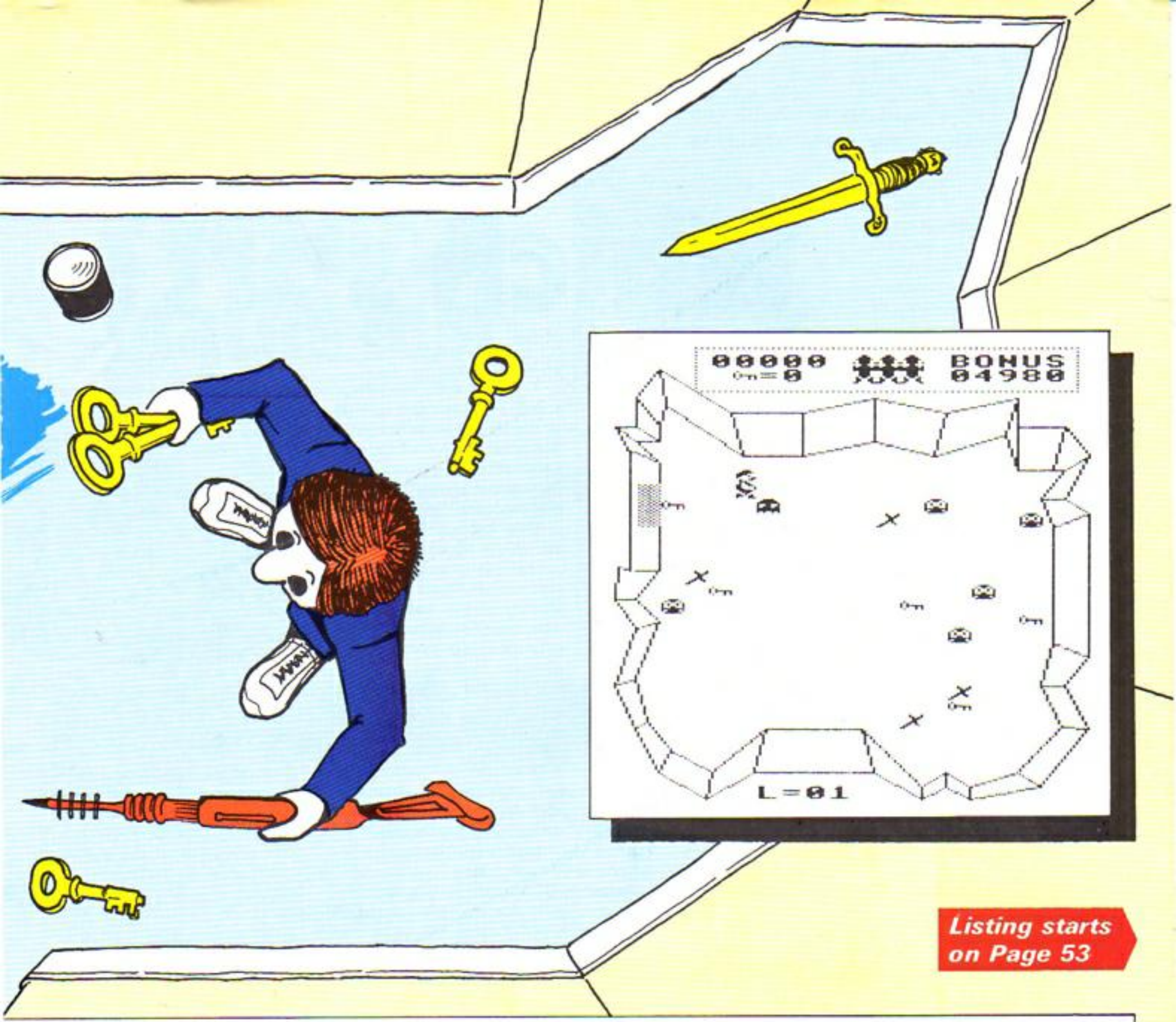
It's weird and it's wonderful. Dare you play Haunted House?

PROCEDURES

PROCinit	Initialise user defined graphics, the main variables and arrays.
PROCassemble	Assemble machine code for moving the aliens.
PROCinstruct	Print the instructions in Mode 4, play a tune and continue.
PROCdrawscreen	Draw the screen, reset variables and start the game.
PROCnewscreen	Move onto the next screen – check for bonus life.
PROCstartgame	Clear keyboard buffer and check the keyboard for starting game.
PROCman	Move the man, check for collisions etc.
PROCgame	Play the game, call the m/c, decrement the bonus and check status.
PROCdead	You have hit an obstacle or an alien – lose a life.
PROCtimeup	Your bonus has run out – lose a life.
PROCfire	Fire your laser if you are facing sideways, check for hits.
PROCleftfire	Draw the laser if you are firing left.
PROCrightfire	Draw the laser if you are facing right.
PROChit	You've hit an alien – increment score and kill it suitably.
PROCalldead	All your lives have run out.
PROCaliens	Place the aliens on the screen.
PROCobjects	A general procedure to place objects on the screen.
PROCw	Wait for a while.
PROCscore	Print the score in the form '00750'.
PROClives	Print the number of lives on the screen.
PROClevel	Print the level in the form '01'.
PROCtune	Play a tune on a defined channel for a defined duration.
PROCinputname	Input your name for the high score table in a certain place.
PROCswap	Swap variables for the high score table.

VARIABLES

x%,y%	Horizontal and vertical position of the man.
man$(4,2)	Various shapes of men for each direction.
d%	Direction of movement, selected for man$().
man%	The particular position of the man's feet.
key%(139)	Ascii values for the key pressed containing subroutine location.
screen%(159)	Ascii values of characters read on the screen, containing line numbers.
dx%(3),dy%(3)	If x% and y% are decremented, value held in this.



Listing starts on Page 53

dx%,dy%	Values for incrementing or decrementing x% and y%.
sc%	Score
li%	Numbers of lives you have left.
lev%	The current level you are on.
bo%	Amount of bonus you have left.
inc%	Increment bonus is reduced by.
doory%	Vertical position of exit door.
mekey%	Number of keys you have collected.
hit%	Number of aliens you have hit.
need%	Number of aliens on the screen for this level.
scr1%,scr2%	C of character read on the screen.
hi%(10)	Hiscores.
hi$(10)	Names of highest scorers.
loop%	General loop variable.

VARIABLES FOR MACHINE CODE

xp%	Located at &70-&7F, the x positions of the aliens.
yp%	Located at &80-&8F, the y positions of the aliens.
di%	Located at &90-&9F, the direction number of the alien.
dead%	Located at &A0-&AF, is the indexed alien dead or not.
scr%	Screen code located at &B0.
rnd%	Random number for use in changing directions, located at &B1.
number%	Number of aliens on the screen, located at &B2.
xc%	The X register is stored in here when testing the screen, held in &B3.
mhit%	If you are hit, it holds a 1 else it holds a 0; located in &B4.
OS	Operating system call, &FFEE.
code_space	Space needed for machine code.
p%	Program counter, set to code_space to provide space for m/c.

SPEED-UP HINTS

- ☐ Use integer variables.
- ☐ Use increments of one for line numbers.
- ☐ Avoid lots of IF . . . THEN comparisons.
- ☐ Use as few spaces as possible.
- ☐ Avoid the use of VDU5 wherever necessary.
- ☐ Use VDU codes instead of COLOUR, GCOL and PRINT TAB commands wherever feasible.
- ☐ Use GOTO and GOSUB variables instead of IF . . . THENs.
- ☐ Place subroutines at the beginning of the program to avoid searching through lines.

What do playing cards, shells and bubbles have to do with computer programming? PAUL HUTSON explains . . .

Which sort?

AS the title suggests, this article sets out to examine the various algorithms available to programmers for sorting information.

Whether this information is numeric or string is irrelevent to Basic, as only the array identity needs to be altered, together with the temporary storage variable, to change from one to the other.

Written in BBC Basic, the example program will run on both the BBC Micro and the Electron, but the timings will differ.

Where variables have been used they've been given names in lower case to help identify their interrelationships.

To aid understanding, it is suggested that one suit from a pack of playing cards should be used to physically work through each routine from the listing.

The first routine that most programmers meet is the "bubble" sort. This is probably because it is the simplest to understand and code and therefore offers the easiest route to early success.

It works by comparing consecutive pairs of elements and swapping any which are not in order. Using one FOR... NEXT loop to step sequentially through the data and another to repeat the process until no more "swaps" take place, it is laborious but easy to follow.

After the first pass the highest item will occupy the top of the list and need not be compared again. Hence the function of the outer FOR... NEXT loop is to reduce the list by one at the end of each pass.

To intercept the situation where the list is ordered before the end of the outer loop, a "flag" has been used. This has no effect when items have been swapped, but brings about an early completion when no swap takes place.

If this routine is RUN four times, and each time the data list is doubled in length, a disproportionate increase in time will be noted.

It is one of the unfortunate side effects of the bubble sort that if the number of items is increased by a factor F then the time taken increases by a factor of F to the power of 2.

Reference to the graph (Figure I) will show that a data list of only 200 items is already taking something like four minutes. This is not too big a problem for a "one off", but will not do for real time situations.

The "interchange" sort is the next algorithm usually encountered, and bears a closer resemblance to real life.

In this routine the list is searched for the lowest (in our case) item and this is placed in position 1. The list is searched again and the next "lowest" item is placed in position 2, and so on.

The power of the FOR... NEXT loop is again utilised, the inner loop to step through the data list and the outer loop to decrease the length of the list on each pass. This avoids comparing items already sorted.

It seems to be a more logical and efficient method that our friendly "bubble" and indeed, as can be seen from the graph, turns out to be significantly faster.

However it still suffers from one major drawback – a time factor which increases exponentially as the data list grows longer.

The "delayed exchange" sort is a streamlined variation of the interchange routine. In this method a new variable, *smallest%*, is used to reduce the number of swaps that take place in each pass.

This offers a significant increase in speed of around 20 per cent. But its operation is a little more obscure and the identity of this extra variable must be changed when using this routine to sort real numbers or strings.

In the search for speed and efficiency those programmers who haven't settled down to the life of a hermit with their pack of playing cards eventually discover the "insertion" sort. Also known as the "sift" sort, this routine bears the closest resemblance to a real life solution.

The method avoids the need for multiple passes, as used in the bubble and interchange routines, and gets the job done in only one pass. Needless to say it is a little more complex in operation but gains greatly in speed over our previous attempts.

The routine steps through our data list comparing consecutive pairs. When a pair is found not to be in order the offending item is placed in a temporary location.

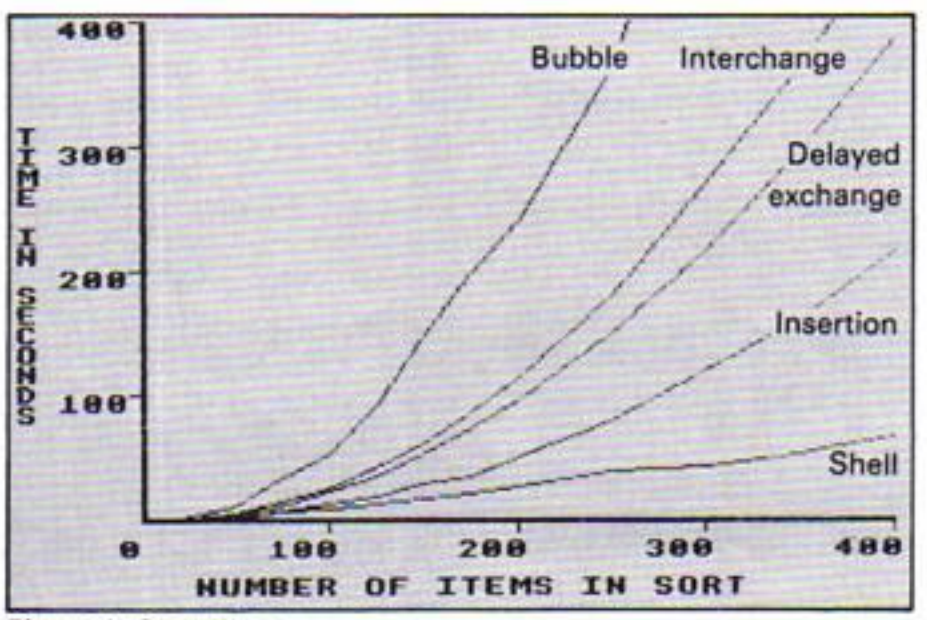


Figure I: Sort times

Now each preceding item is moved along the list until the correct position for our temporary item opens up, whereupon it is reinserted into the list.

Whether this is really more difficult to understand than our earlier attempts is open to debate, but it does require more thought to code.

However we still find that it suffers from the same drawback mentioned earlier, a time penalty exponentially increasing with each increase in the length of our data list.

We've already maximised the ever faithful FOR . . . NEXT loop. What's needed is a method of swapping items over long distances to minimise the number of times each item has to be moved. This brings us to the elusive "shell" sort – named after its originator, D. Shell.

Having referred to the comparative graph several times already it will have been observed that, while not being particularly stunning when sorting up to 150 items, the shell sort offers very great savings in time when sorting 200 plus items.

Also obvious is the fact that this method does not suffer intolerable increases in sort time when the data list grows longer.

In fact for very long data lists it will out-perform a machine code sort which utilises one of the other methods. A typical machine code bubble sort will take 800 seconds to sort 4,000 items, our Basic shell sort can manage it in around 640 seconds – but a Basic bubble sort would take 24 hours to do the same job!

However a glance at the code reveals a rather more intricate programming task.

If compared to a chain, our data list is divided into two "links". The first data item in each link is compared for order

and sorted if necessary. The second data item in each link is then operated on in the same way and so on to the last item in each link.

Now the link size is divided by two and the operation repeated. Successive halving of the link size eventually brings it to a value of 1, at which time the whole list is sorted together.

At first sight it's difficult to see how such a long winded routine can be efficient. Many short sorts are performed which exchange items over large distances. This reduces the number of times items have to be moved.

Also, as can be seen from the graph, it is much quicker to sort 10 lists of 10 items than one list of 100 items.

The powerhouse for this routine is in fact an insertion sort, the shell side of things being to present the data list in small chunks.

This does involve rather more coding effort and only shows its efficiency on long lists.

However this method has a major advantage when used on long data lists when, if the data list is doubled in length, the time taken only increases by a factor of about 2.4.

When you've convinced yourself that you understand how each method works, try altering the routines to sort data in the reverse order.

This isn't always as simple as might be imagined and it's worth trying it with our faithful playing cards before altering the code.

For those who need a faster method of sorting there is now only one place left to go – machine code. This is so fast as to make sorting in many situations totally transparent to the user.

This, however, will be the subject of a future article . . . if I ever stop playing with these cards.

```
  10 REM      :::::WHICH S
ORT:::::
  20 REM BY PAUL HUTSON
  30 REM (C) ELECTRON USER
  40 X=RND(-TIME)
  50 DIM S%(400)
  60 FOR Num=25 TO 400 STE
P25
  70   PRINT''"Num=";Num
  80   PROCrandomise:PROCb
ubble
  90   PROCrandomise:PROCi
nterchange
 100   PROCrandomise:PROCd
elayedexchange
 110   PROCrandomise:PROCi
nsertion
 120   PROCrandomise:PROCs
hell
 130   NEXT Num
 140 END
 150
 160
 170 DEFPROCbubble
 180 TIME=0
 190 FOR index1=Num-1 TO 1
 STEP -1
 200   LET flag=-1
 210   FOR index2=1 TO ind
ex1
 220     IF S%(index2)>S%
(index2+1) THEN LET temp%=S
%(index2):LET S%(index2)=S%
(index2+1):LET S%(index2+1)
=temp%:LET flag=0
 230     NEXT index2
 240   IF flag=-1 THEN LET
 index1=1
 250   NEXT index1
 260 PRINT"BUBBLE SORT TOO
K ";TIME/100;" seconds"
 270 ENDPROC
 280
 290
 300 DEF PROCinterchange
 310 TIME=0
 320  FOR Index1=1 TO Num-
1
 330    FOR Index2=Index1+
1 TO Num
 340     IF S%(Index2)>=S%
(Index1) THEN 380
 350     temp%=S%(Index2)
 360     S%(Index2)=S%(Ind
ex1)
 370     S%(Index1)=temp%
 380     NEXT Index2
 390   NEXT Index1
 400 PRINT"INTERCHANGE SOR
T TOOK ";TIME/100;" seconds
"
 410 ENDPROC
 420
 430
 440 DEF PROCinsertion
 450 TIME=0
 460  FOR index1=2 TO Num
 470    temp%=S%(index1)
 480   FOR index2=index1 T
O 2 STEP-1
 490     S%(index2)=S%(ind
ex2-1)
 500     IF temp%>=S%(inde
x2) THEN S%(index2)=temp%:i
ndex2=2
 510     NEXTindex2
 520   IF temp%<S%(1) THEN
 S%(1)=temp%
 530   NEXT index1:PRINT"I
NSERTION SORT TOOK ";TIME/1
00;" seconds"
 540 ENDPROC
 550
 560
 570 DEF PROCshell
 580 TIME=0
 590 LET link=Num
 600 REPEAT
 610   LET link=INT(link/2
)
 620   FOR index2=1 TO lin
k
 630     linkstep=index2+l
ink
 640     FOR index3=linkst
ep TO Num STEP link
 650       LET temp%=S%(in
dex3)
 660       FOR index4=inde
x3 TO linkstep STEP -link
 670         LET S%(index4
)=S%(index4-link)
 680         IF temp%>=S%
(index4) THEN LET S%(index4
)=temp%:LET index4=linkstep
 690         NEXTindex4
 700       IF temp%<S%(ind
ex2) THEN LET S%(index2)=te
mp%
 710       NEXTindex3
 720     NEXTindex2
 730   UNTIL link=1
 740 PRINT"SHELL SORT TOOK
 ";TIME/100;" seconds"
 750 ENDPROC
 760
 770
 780 DEF PROCrandomise
 790 FOR X=0 TO Num:S%(X)=
RND(Num)
 800   NEXTX
 810 ENDPROC
 820 DEF PROCdelayedexchan
ge
 830 TIME=0
 840 FOR Index1=1 TO Num-1
 850  smallest%=Index1
 860   FOR Index2=Index1+1
 TO Num
 870    IF S%(Index2)>=S%(
smallest%) THEN 890
 880    smallest%=Index2
 890    NEXT Index2
 900    temp%=S%(smallest%
)
 910    S%(smallest%)=S%(I
ndex1)
 920    S%(Index1)=temp%
 930   NEXTIndex1
 940  PRINT"DELAYED EXCHAN
GE SORT TOOK ";TIME/100;"se
conds"
 950 ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

AS we've seen from Paul Hutson's article, there's more than one sorting algorithm. ROLAND WADDILOVE's program has five of them at work sorting words into an alphabetical list. By showing each step of the sort it makes the processes a little clearer. You could say that it sorts things out!

Time you got yourself sorted out

PROCEDURES

PROCbubble_sort
PROCexchange_sort
PROCdel_rep_sort
PROCShell_Metzner
PROCsift_sort
— The sorting algorithms

PROCinitialise Sets up the two arrays needed, switches off the cursor keys, Escape and the auto repeat. Defines the function keys, calls PROCget_list to read in the list of words.

PROCnotes Prints a few notes about the program.

PROCset_list Copies the list of words into another array so that it can be sorted. Sets the number of swaps and comparisons to zero.

PROCmenu Prints the menu.

PROCkey( ) Restricts input to the keys shown.

PROCinput_list Allows you to type in a new list of words.

PROClist Prints the final page.

PROCprint_list Prints the list of words to be sorted.

PROCcompare( ) Highlights the words being compared.

PROCswap( ) Swaps the words around, calls PROCmove( ) to swap the words on the screen.

VARIABLES

word$( ) Holds the words to be sorted.

list$( ) The words are copied into this and sorted.

end End of the list – the number of words.

i,j,k,n,m,l,yn General variables, used for many things.

swaps How many words have been swapped around.

comparisons How many words have been compared.

temporary$ Used to store a word when swapping two items.

```
  10REM **      SORTS          list
  **                             170UNTIL INSTR("Ee",key$)
  20REM ** By R.A.Waddilov       180MODE 6
e **                             190PROClist
  30                             200END
  40MODE 1                       210
  50PROCinitialise               220DEF PROCsift_sort
  60PROCnotes                    230PROCprint_list
  70REPEAT                       240FOR i=2 TO end
  80PROCset_list                 250PROCcompare(i,i-1)
  90PROCmenu                     260IF list$(i) < list$(i-
 100PROCkey("123456Ee")      1)                  THEN PR
 110IF key$="1" PROCbubble   OCshift(i)
_sort                            270NEXT i
 120IF key$="2" PROCexchan       280PROCfinished
ge_sort                          290ENDPROC
 130IF key$="3" PROCdel_re       300
p_sort                           310DEF PROCshift(i)
 140IF key$="4" PROCShell_       320temporary$=list$(i)
Metzner                          330j=i-1 : done=FALSE
 150IF key$="5" PROCsift_s       340REPEAT swaps=swaps+1
ort                              350list$(j+1)=list$(j)
 160IF key$="6" PROCinput_       360IF j=1 THEN done=TRUE
                                                 ELSE PR
                             OCcompare(j-1,j-1) :
                                  IF temporary$ > list$
                             (j-1)               THEN d
                             one=TRUE ELSE j=j-1
                                 370UNTIL done
                                 380list$(j)=temporary$
                                 390PROCprint_list : PROCp
                             ause(100)
                                 400ENDPROC
                                 410
                                 420DEF PROCShell_Metzner
                                 430PROCprint_list
                                 440i=0
                                 450REPEAT i=i+1
                                 460UNTIL 2^i > end
                                 470j=2^i-1
                                 480REPEAT j=INT(j/2)
                                 490k=end-j : a=1
                                 500REPEAT i=a
                                 510REPEAT b=i+j
                                 520PROCcompare(i,b)
                                 530IF list$(i)>list$(b)
                                                 THEN PR
                             OCswap(i,b) :
                                      i=i-j : done=i<1
                                                 ELSE don
                             e=TRUE
                                 540UNTIL done
                                 550a=a+1
                                 560UNTIL a>k
                                 570UNTIL j=1
                                 580PROCfinished
                                 590ENDPROC
                                 600
                                 610DEF PROCexchange_sort
                                 620PROCprint_list
                                 630FOR j=1 TO end-1
                                 640FOR i=j+1 TO end
                                 650PROCcompare(i,j)
                                 660IF list$(i)<list$(j)
                                                 THEN PR
                             OCswap(i,j)
                                 670NEXT i
                                 680NEXT j
                                 690PROCfinished
```

```
 700ENDPROC
 710
 720DEF PROCdel_rep_sort
 730PROCprint_list
 740FOR j=1 TO end-1
 750smallest=j
 760FOR i=j+1 TO end
 770PROCcompare(i,smallest
)
 780IF list$(i)<list$(smal
lest)              THEN sm
allest=i
 790NEXT i
 800PROCswap(j,smallest)
 810NEXT j
 820PROCfinished
 830ENDPROC
 840
 850DEF PROCbubble_sort
 860PROCprint_list
 870FOR j=end TO 2 STEP -1
 880FOR i=2 TO j
 890PROCcompare(i,i-1)
 900IF list$(i)<list$(i-1)
                  THEN PR
OCswap(i,i-1)
 910NEXT i
 920NEXT j
 930PROCfinished
 940ENDPROC
 950
 960DEF PROCswap(n,m)
 970IF n=m ENDPROC
 980swaps=swaps+1
 990PRINT TAB(30,28);swaps
1000PROCmove(n,m)
1010temporary$=list$(n)
1020list$(n)=list$(m)
1030list$(m)=temporary$
1040ENDPROC
1050DEF PROCinitialise
1060DIM word$(10),list$(10
)
1070end=10:PROCget_list
1080*FX4,1
1090*FX11,0
1100*FX229,1
1110*KEY0,"RUN !M"
1120*KEY1,"LIST07!MLIST850
,940!L!M"
1130*KEY2,"LIST07!MLIST610
,700!L!M"
1140*KEY3,"LIST07!MLIST720
,830!L!M"
1150*KEY4,"LIST07!MLIST420
,590!L!M"
1160*KEY5,"LIST07!MLIST220
,400!L!M"
1170*KEY6,"PROClist!L!M"
1180*KEY10,"OLD!MRUN!M"
1190ENDPROC
1200DEF PROCget_list
1210FOR i=1 TO 10
1220READ word$(i)
1230NEXT i
1240ENDPROC
1250DATA dog,cat,mouse,ele
phant,horse,goldfish,hamste
r,sheep,goat,kangaroo
1260DEF PROCset_list
1270FOR i=1 TO 10
1280list$(i)=word$(i)
1290NEXT i
1300comparisons=0:swaps=0
1310ENDPROC
1320DEF PROCcompare(n,m)
1330comparisons=comparison
s+1
1340PRINT TAB(14,28);compa
risons
1350COLOUR129:COLOUR2:PRIN
T TAB(15,3+n*2);list$(n);TA
B(15,3+m*2);list$(m)
1360PROCpause(100)
1370COLOUR128:COLOUR3:PRIN
T TAB(15,3+n*2);list$(n);TA
B(15,3+m*2);list$(m)
1380PROCpause(50)
1390ENDPROC
1400DEF PROCprint_list
1410COLOUR3:CLS
1420FOR i=1 TO 10
1430PRINT TAB(15,3+i*2);li
st$(i)
1440NEXT i
1450COLOUR2:PRINT TAB(2,28
);"Comparisons:";comparison
s;" ";TAB(22);"Swaps:";swap
s;" "
1460ON VAL key$ GOTO 1470,
1480,1490,1500,1510
1470PRINT TAB(13,1);"BUBBL
E SORT":ENDPROC
1480PRINT TAB(12,1);"EXCHA
NGE SORT" : ENDPROC
1490PRINT TAB(7,1);"DELAYE
D REPLACEMENT SORT":ENDPROC
1500PRINT TAB(9,1);"SHELL-
METZNER SORT":ENDPROC
1510PRINT TAB(14,1);"SIFT
SORT":ENDPROC
1520
1530DEF PROCmove(n,m)
1540COLOUR 1
1550FOR I=0 TO 12
1560PRINT TAB(15-I,3+n*2);
list$(n);" ";TAB(14+I,3+m*2
);" ";list$(m)
1570PROCpause(5)
1580NEXT I
1590yn=3+n*2:ym=3+m*2
1600REPEAT
1610PRINT TAB(3,yn);list$(
n);TAB(27,ym);list$(m)
1620PROCpause(20)
1630PRINT TAB(3,yn);SPC(10
);TAB(27,ym);SPC(10)
1640yn=yn+(yn>3+m*2)-(yn<3
+m*2)
1650ym=ym+(ym>3+n*2)-(ym<3
+n*2)
1660UNTIL yn=3+m*2
1670FOR I=0 TO 12
1680PRINT TAB(2+I,3+m*2);"
";list$(n);TAB(27-I,3+n*2)
;list$(m);" "
1690PROCpause(5)
1700NEXT I
1710COLOUR3:PRINT TAB(15,3
+n*2);list$(m);TAB(15,3+m*2
);list$(n)
1720PROCpause(50)
1730ENDPROC
1740DEF PROCfinished
1750COLOUR1:PRINT TAB(2,30
);"Finished - press the spa
ce bar...";
1760PROCkey(" ")
1770ENDPROC
1780DEF PROClist
1790*FX229,0
1800*FX4,0
1810*FX12,0
1820VDU 19,1,3,0,0,0
1830PRINT'" ** FUNCTION K
EYS **"'" f0. Run the pro
gram again."'" f1. List Bu
bble sort."'" f2. List Exc
hange sort."'" f3. List De
layed Replacement sort."
1840PRINT'" f4. List Shell
-Metzner sort."'" f5. List
 Sift sort."'" f6. Return
to this page."'''" On the E
lectron the function keys a
re  obtained by holding do
wn CAPS LK/FUNC  and press
ing a number ";
1850ENDPROC
1860DEF PROCmenu
1870COLOUR3:CLS:PRINT''TAB
(5);"MENU"'TAB(4);"------"'
1880COLOUR2:PRINT'" 1. Bub
ble Sort."'" 2. Exchange S
ort."'" 3. Delayed Replace
ment Sort."'" 4. Shell-Met
zner Sort."'" 5. Sift Sort
."'" 6. Input new list of
words."''" E. End program
and List methods."
1890COLOUR1:PRINT'''" Pres
s a key...";
1900ENDPROC
1910DEF PROCkey(allowed$)
1920VDU 23,1,1;0;0;0;
1930*FX15,1
1940REPEAT key$=GET$
1950UNTIL INSTR(allowed$,k
ey$)
1960VDU 23,1,0;0;0;0;
1970ENDPROC
1980DEF PROCpause(delay)
1990TIME=0
2000REPEAT
2010UNTIL TIME > delay
2020ENDPROC
2030DEF PROCinput_list
2040CLS:COLOUR3:PRINT''"Th
is program is only a demons
tration of"''"how sorting p
rograms work.":COLOUR2:PRIN
T''"There must be a list of
 ten words with"''"a maximu
m of ten letters."
2050COLOUR1:PRINT''"The wo
rds must be all UPPER CASE
or all"''"lower case."
2060VDU 17,3,23,1,1;0;0;0;
2070FOR i=1 TO 10
2080PRINT TAB(0,17);"Word:
";i
2090REPEAT
2100INPUT TAB(0,20);SPC(80
);TAB(0,20);word$(i)
2110UNTIL LEN word$(i)<11
AND LEN word$(i)
2120SOUND 1,-10,100,5
2130NEXT i
2140ENDPROC
2150DEF PROCnotes
2160PRINT'TAB(15);"SORTS"'
TAB(14);"-------"
2170COLOUR2:PRINT''"The obj
ect of this program is to s
how"''"how a list of words
can be sorted into"''"alpha
betical order by the comput
er."
2180PRINT''"There is a cho
ice of five different"''"me
thods and the option of inp
utting your""own list of w
ords."
2190PRINT''"The words bein
g compared are highlighted"
''"and, ( except for the sif
t sort method"''"which work
s in a slightly different w
ay)""the words being swapp
ed round are shown."
2200COLOUR1:PRINT''TAB(11
);"Press space...";
2210PROCkey(" ")
2220ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

IT was with great interest that I read Nigel Peter's introduction to multi-coloured user defined graphics in the June 1984 *Electron User,* as my class and I were then involved in an identical activity.

Actually we'd begun rather differently – with the binary system – but soon strayed to the idea of a computer using binary notation. I explained how a figure on screen was actually a series of blocks within an 8 x 8 grid.

The method of creating these, using a VDU 23 command, can be studied in the User Guide, pages 93-95.

Of course, children being children, they soon developed their ideas in different ways, and I offered to bring my Electron into school so that we could put their designs on screen.

While some were content with the limitations of an 8 x 8 grid, many realised that much better results came from a larger shape.

Others soon wanted to put more than just two colours, background and foreground, into their figures, and so I had to use colour overlays to achieve a likeness on screen of their designs.

Readers unfamiliar with VDU codes may not realise that PRINT CHR$ 235 may be replaced by the shorter and easier statement VDU 235.

A second and greater advantage is gained by the fact that VDU statements may be strung together with commas, except when the syntax demands a semi-colon.

For instance, the command to print characters 235-239 all at the same place may be easily written as VDU 235,8,236,8,237,8,238,8, 239 (VDU 8 moves the cursor one space left).

The trouble is that each subsequent character erases the previous one, but the answer is simple and is yet another VDU command.

VDU 5 combines the text and graphics cursors and allows figures to be superimposed upon each other, so that "layers" of colour might be built up with previous layers showing through.

In the listing I have stuck to the GCOL 0,n statement to define colours, but I could have used yet another VDU statement, so that

```
230 GCOL 0,1:VDU228,231
```

would become

```
230 VDU18,0,1,228,231
```

but there comes a point where readability for debugging becomes an asset!

The rest of the program concerns nested FOR ... NEXT loops so that the designs might be shown as a full pattern on the screen.

It might also be useful to note that these were produced in colour on a good quality monitor.

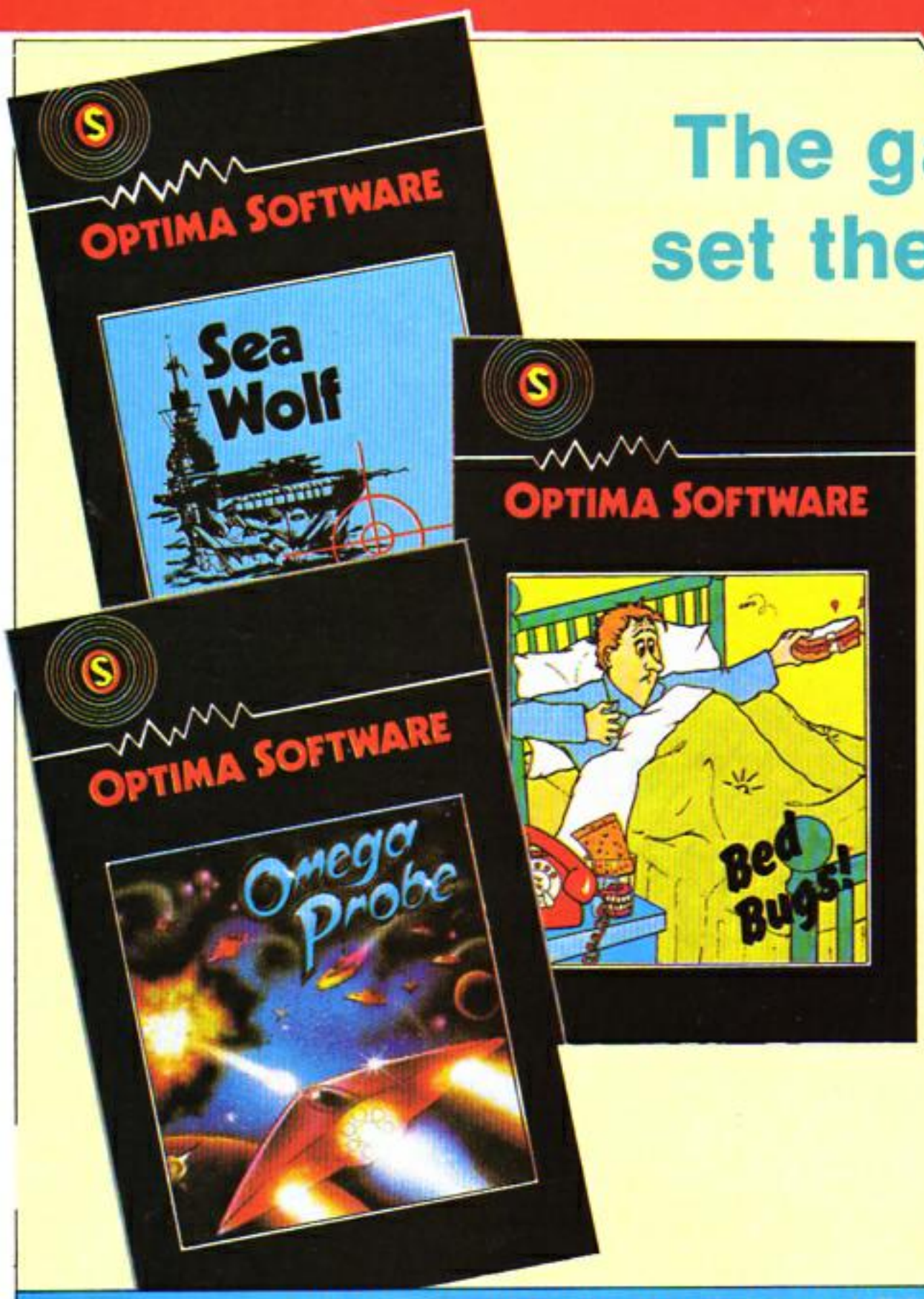
A domestic TV does not show the designs to their best advantage, but produce an interesting sideline . . . the butterfly's wings may well appear orange on a television, a colour which is impossible on an Electron.

```
  10 REM ******************
********
  20 REM ** COLOURED CHARA
CTERS **
  30 REM ******************
********
  40 REM ****** PHIL TAYLE
R ******
  50 REM *** (C) ELECTRON
USER ***
  60 REM ******************
********
  70
  80 REPEAT
  90
 100 REM ***PAUL'S TRAIN *
**
 110VDU23,228,0,0,0,0,253,
0,0,0
 120VDU23,229,0,0,0,1,0,25
3,0,0
 130VDU23,230,0,0,0,0,0,2,
221,136
 140VDU23,231,0,0,0,0,254,
0,0,0
 150VDU23,232,0,0,4,132,0,
126,0,0
 160VDU23,233,48,8,0,16,0,
128,221,136
 170MODE2
 180 VDU5
 190 GCOL0,133:CLG
 200 FOR Y=100 TO 1000 STE
P 100
 210 FOR X=100 TO 1200 STE
P 200
 220 MOVE X,Y
 230GCOL0,1:VDU228,231
 240GCOL0,2:VDU8,8,229,232
 250GCOL0,0:VDU8,8,230,233
 260 NEXT X
 270 NEXT Y
 280 PROCdelay
 290
 300REM *** EMMA'S CLOWN *
**
 310VDU23,234,48,48,0,0,0,
1,0,0
 320VDU23,235,1,1,1,7,0,0,
0,0
 330VDU23,236,0,0,0,0,1,0,
0,0
 340VDU23,237,0,0,8,0,0,0,
0,0
 350VDU23,238,0,0,0,0,2,2,
3,0
 360VDU23,239,0,0,0,0,0,0,
0,16
 370VDU23,240,0,0,0,0,0,64
,128,0
 380VDU23,241,192,192,192,
240,0,0,0,0
 390VDU23,242,0,0,0,0,64,0
,0,0
 400VDU23,243,0,0,0,0,160,
32,96,128
 410VDU23,244,0,0,0,0,0,12
8,0,0
 420VDU23,245,0,0,0,0,0,0,
0,4
 430VDU23,246,9,7,0,3,6,0,
0,0
 440VDU23,247,0,0,1,0,0,60
,60,0
 450VDU23,248,0,0,0,0,0,0,
0,62
 460VDU23,249,200,240,0,19
2,96,0,0,0
 470VDU23,250,0,0,192,0,0,
120,120,0
 480VDU23,251,0,0,0,0,0,0,
0,120
 490 GCOL0,134: CLG
 500FOR Y=100 TO 1000 STEP
200
 510FOR X=100 TO 1200 STEP
200
 520MOVE X,Y
```

```
  530GCOL0,1:VDU234,240
  540GCOL0,0:VDU8,8,235,241
  550GCOL0,4:VDU8,8,236,242
  560GCOL0,2:VDU8,8,237,244
,10,8,8,248,251
  570GCOL0,3:VDU8,8,11,238,
243,10,8,8,247,250
  580GCOL0,5:VDU11,8,8,239,
245,10,8,8,246,249
  590NEXT X
  600NEXT Y
  610 PROCdelay
  620
  630 REM ***SARAH'S BUTTER
FLY***
  640VDU23,224,0,8,4,2,1,1,
1,0
  650VDU23,225,0,0,0,16,32,
24,40,21
  660VDU23,226,0,0,0,32,88,
40,86,42
  670VDU23,227,0,16,32,64,1
28,128,128,0
  680VDU23,228,0,0,0,8,4,40
,20,168
  690VDU23,229,0,0,0,4,26,2
0,106,84
  700VDU23,230,0,1,1,1,1,0,
0,0
  710VDU23,231,84,40,84,32,
80,0,0,0
  720VDU23,232,42,22,40,24,
32,0,0,0
  730VDU23,233,0,128,128,12
8,128,0,0,0
  740VDU23,234,170,20,42,4,
10,0,0,0
  750VDU23,235,84,104,20,24
,4,0,0,0
  760 GCOL0,128:CLG
  770 FOR Y=100 TO 1000 STE
P 200
  780  FOR X=100 TO 1200 ST
EP 200
  790MOVEX,Y
  800GCOL0,3:VDU224,227,10,
8,8,230,233
  810GCOL0,3:VDU8,8,11,225,
228,10,8,8,232,235
  820GCOL0,1:VDU8,8,11,226,
229,10,8,8,231,234
  830 NEXT X
  840 NEXT Y
  850 PROCdelay
  860
  870 REM *** KATE'S MAN **
*
  880VDU23,224,0,0,127,97,9
7,126,126,127
  890VDU23,225,0,0,0,30,24,
0,0,0
  900VDU23,226,0,0,0,0,6,0,
0,0
  910VDU23,227,0,0,0,0,0,1,
1,0
  920VDU23,228,0,0,254,134,
134,126,126,254
  930VDU23,229,0,0,0,126,12
,0,0,0
  940VDU23,230,0,0,0,0,96,0
,0,0
  950VDU23,231,0,0,0,0,0,12
8,128,0
  960VDU23,232,127,127,127,
112,112,127,0,0
  970VDU23,233,0,0,0,10,5,0
,0,0
  980VDU23,234,254,254,254,
14,14,254,0,0
  990VDU23,235,0,0,0,160,80
,0,0,0
 1000CLG
 1010FOR Y=100 TO 1000 STEP
200
 1020FOR X=100 TO 1200 STEP
200
 1030MOVE X,Y
 1040GCOL0,3:VDU224,228,8,8
,10,232,234
 1050GCOL0,4:VDU8,8,11,225,
229
 1060 GCOL0,1:VDU8,8,226,23
0
 10706COL0,2:VDU8,8,227,231
 1080 GCOL0,0:VDU8,8,10,233
,235
 1090 NEXT X
 1100 NEXT Y
 1110 PROCdelay
 1120
 1130 REM *** rICHARD'S POR
TRAIT ***
 1140VDU23,224,0,0,1,0,0,0,
0,0
 1150VDU23,225,0,0,0,3,0,4,
0,2
 1160VDU23,226,0,0,0,0,0,1,
0,0
 1170VDU23,227,0,0,0,0,15,0
,0,0
 1180VDU23,228,0,0,128,0,0,
0,0,0
 1190VDU23,229,0,0,0,192,0,
32,0,64
 1200VDU23,230,0,0,0,0,0,12
8,0,0
 1210VDU23,231,0,0,0,0,240,
0,0,0
 1220VDU23,232,128,0,0,0,0,
0,0,0
 1230VDU23,233,0,0,0,240,24
0,144,144,240
 1240VDU23,234,0,0,0,15,15,
32,64,0
 1250VDU23,235,0,0,0,0,0,64
,32,0
 1260VDU23,236,1,0,0,0,0,0,
0,0
 1270VDU23,237,0,0,0,15,15,
15,15,15
 1280VDU23,238,0,0,0,240,24
0,0,0,0
 12906COL0,134:CLG
 1300FOR Y=100 TO 1000 STEP
200
 1310FOR X=100 TO 1200 STEP
200
 1320MOVE X,Y
 1330GCOL0,4:VDU224,228
 1340GCOL0,1:VDU8,8,225,229
,8,8,10,232,236
 1350GCOL0,2:VDU8,8,11,226,
230,8,8,10,235
 1360GCOL0,5:VDU11,8,227,23
1,8,8,10,233,237
 1370GCOL0,3:VDU8,8,234,238
 1380NEXT X
 1390NEXT Y
 1400 PROCdelay
 1410
 1420 REM *** CATHERINE'S M
ETAL MICKEY ***
 1430VDU23,224,6,0,112,64,1
13,0,15,15
 1440VDU23,225,0,15,9,15,12
,1,0,0
 1450VDU23,226,0,0,2,0,2,0,
0,0
 1460VDU23,227,0,0,4,48,0,0
,0,0
 1470VDU23,228,96,0,0,0,128
,0,240,240
 1480VDU23,229,0,240,144,24
0,48,128,0,0
 1490VDU23,230,0,0,64,0,64,
0,0,0
 1500VDU23,231,0,0,32,0,0,0
,0,0
 1510VDU23,232,15,15,12,12,
0,0,0,0
 1520VDU23,233,0,0,1,0,0,0,
0,0
 1530VDU23,234,0,0,0,0,12,0
,0,0
 1540VDU23,235,0,0,2,3,3,3,
3,15
 1550VDU23,236,240,240,48,4
8,0,0,0,0
 1560VDU23,237,0,0,128,0,0,
0,0,0
 1570VDU23,238,0,0,0,0,48,0
,0,0
 1580VDU23,239,0,0,64,192,1
92,192,192,240
 1590 GCOL0,128:CLG
 1600FOR Y=100TO1000 STEP20
0
 1610FOR X=100TO1200 STEP20
0
 1620MOVE X,Y
 1630GCOL0,3:VDU224,228,8,8
,10,232,236
 1640GCOL0,1:VDU8,8,11,225,
229
 1650GCOL0,4:VDU8,8,226,230
,8,8,10,233,237
 1660GCOL0,2:VDU8,8,11,227,
231,8,8,10,235,239
 1670GCOL0,0:VDU8,8,234,238
 1680NEXT X
 1690NEXT Y
 1700 PROCdelay
 1710
 1720 REM ** BENN'S FRANKEN
STEIN **
 1730VDU23,224,0,15,6,0,2,0
,0,0
 1740VDU23,225,0,0,10,15,9,
31,15,15
 1750VDU23,226,0,240,80,0,3
2,0,0,0
 1760VDU23,227,0,0,160,240,
144,248,240,240
 1770VDU23,228,0,0,7,7,0,0,
0,0
 1780VDU23,229,15,15,8,8,15
,15,15,15
 1790VDU23,230,0,0,0,0,0,0,
24,0
 1800VDU23,231,0,0,224,224,
0,0,0,0
 1810VDU23,232,240,240,16,1
6,240,240,240,240
 1820VDU23,233,0,0,0,0,0,0,
24,0
 1830 GCOL0,133:CLG
 1840FOR Y=100 TO 1000 STEP
200
 1850FOR X=100 TO 1200 STEP
200
 1860MOVE X,Y
 1870GCOL0,2:VDU224,226,8,8
,10,228,231
 1880GCOL0,1:VDU8,8,11,225,
227,8,8,10,229,232
 1890GCOL0,0:VDU8,8,230,233
 1900NEXT X
 1910NEXT Y
 1920PROCdelay
 1930 UNTIL FALSE
 1940 END
 1950 DEFPROCdelay
 1960 FOR N=1TO5000:NEXT
 1970 ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

OPTIMA SOFTWARE



The games that set the standard

SEA WOLF

So far all has gone well. You have successfully guided your submarine safely through enemy controlled waters and you are beginning to relax.

Suddenly alarm bells scream in your ears – you are under attack!

Desperately you scan the radar screen. Should you try to get him within range of your torpedoes, or attempt evasive tactics? Can you lead your crew to safety?

BED BUGS

The pests are after your feet and you'll have to move fast to stop them. Swot them with a jam sandwich or crunch them with your false teeth.

If you're desperate you can always phone for help. But whatever you do, do it quickly. You need cunning tactics and nimble fingers!

Bed Bugs guarantees hours of hilarity for the whole family.

OMEGA PROBE

Far out in the uncharted reaches of the universe lie the Omega zones from which no man has ever returned.

To explore this hazardous region the Earth's scientists have created the Omega Probe – the ultimate spacecraft.

As pilot of the probe, you face the unknown hazards of the Omega zones. Your mission: to survive.

This fast and furious machine code game with its tremendous graphics and many unique features takes arcade games to new heights of programming excellence.

Get these great games from your Acorn dealer or send off the coupon below to:
Optima Software Ltd, 36 St. Petersgate, Stockport SK1 1HL.

Sea Wolf
- ☐ BBC 'B' cassette £6.95
- ☑ Electron cassette £6.95
- ☐ BBC 40 track disc £8.95
- ☐ BBC 80 track disc £8.95

Bed Bugs
- ☐ BBC 'B' cassette £6.95
- ☐ Electron cassette £6.95
- ☐ BBC 40 track disc £8.95
- ☐ BBC 80 track disc £8.95

Omega Probe
- ☐ BBC 'B' cassette £6.95
- ☐ Electron cassette £6.95
- ☐ BBC 40 track disc £8.95
- ☐ BBC 80 track disc £8.95

☐ I enclose acheque payable to Optima Software Ltd.

☐ I wish to pay by *Access/Visa (*delete as appropriate).

Name _____ Card No. _____

Address _____ Expiry date _____

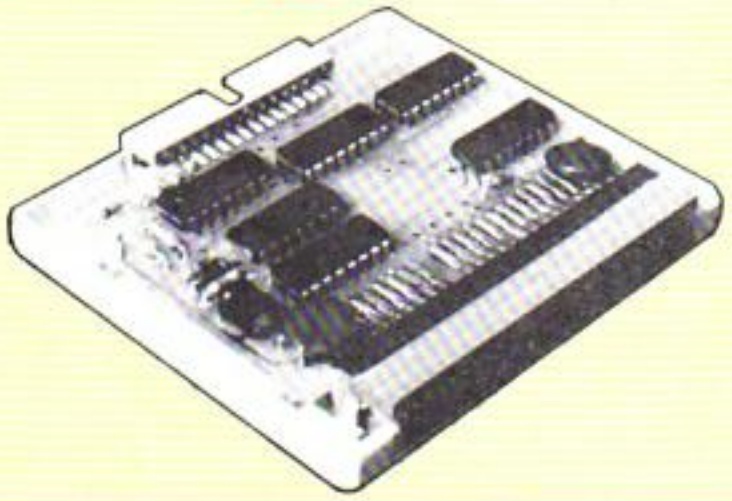
_____

_____ Signed _____

WIN

this great new printer interface from First Byte in our fantastic FREE contest

Spot the listing difference



THIS month we've got a competition with a difference. In fact it's a spot-the-difference competition, and the two lucky winners will each receive the new First Byte printer interface.

Take a look at the two listings below:

As you can see, the gremlins have gone to work again. Program II (which doesn't work) is supposed to be the same as Program I (which does work).

Your job is to find the differences. When you think you've got them all just tell us how many you've found, complete the sentence on the coupon below (in not more than 20 words) and send it to us.

You could win one of two First Byte printer interfaces. Entries close on September 30. The judge's decision is final.

```
 10 REM PROGRAM I
 20 MODE 1
 30 GCOL 3,1
 40 FOR X=0 TO 500 STEP 3
2
 50 MOVE 500+X,500
 60 DRAW 500,1000-X
 70 DRAW 500-X,500
 80 DRAW 500,X
 90 DRAW 500+X,500
100 NEXT
```

Program I

```
 10 REM PROGRAM II
 20 MODE 1
 30 GCOL 3,1
 40 FOR x=0 TO 500 STEP 3
2
 50 MOVE 500+X,500
 60 DRAW 500,1000+X
 70 move 500-X;500
 80 DRAW 5000,X
 90 DRAW 500*X,5000
100 NEXT Y
```

Program II

JUNE CONTEST WINNER

REMEMBER the June competition? We'd got our listings in a twist and asked you to sort it out for us.

The number of entries we got was staggering, you must have been ploughing through the back issues for hours.

As a tie breaker we asked you to tell us the feature that you'd most like to see in *Electron User*. Again the response was enormous and we've learnt a lot.

Future issues of the magazine will reflect your views.

The answer, for those who didn't manage the search through the back issues, was pages 4, 12, 1, 29, 30, 13, 53 and 57.

The winner was T. POOL of Glasgow who not only got the pages right but also won the favour of the judges with his tie-breaker.

As he said: "The feature that I would most like to see in *Electron User* is the one announcing that I have been the lucky winner of June's competition".

The Mushroom printer and user-port interface is on its way.

Your FREE Electron User Contest entry form

I found _____ differences.

I need a printer interface because _____

_____

_____

_____

Name _____

Address _____

_____

_____

Send to: GREMLINS, Electron User Contest, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

NIGEL PETERS looks into ways of using the keyboard

Make sure those are under your

GRAPHIC Control by Ian Rodgers is the program that comes under the magnifying glass this month. It shows how you can control the movement of a user defined character across the screen by means of the keyboard.

It is simple but effective, and demonstrates one of the basic games techniques.

The first three lines of the program are just the usual REM statements. These tell us its name, who wrote it and that it involves the cursor keys. These are the ones with the arrows on them that you'll find at the top right of the keyboard.

Line 40 puts the Electron into Mode 1 (you'd guessed that bit, hadn't you?), a four colour mode with 32 lines, each line having 40 characters. As you probably know by now, line 50 switches off the flashing cursor.

It's amazing the number of otherwise excellent programs coming into the *Electron User* office which are spoilt by an ugly cursor leaping merrily across the screen. The VDU23 of line 50 suppresses the brute.

Line 60 is another VDU23, this time one which defines the little alien character, shown in Figure I. In place of the alien you could have any of the Casting Agency characters hurtling about the screen.

Lines 70 and 80 set the variables *X* and *Y* to zero. Since later in the program *X* and *Y* are used as the X and Y coordinates of the TAB statement that prints the little alien, this means that he starts out sitting at the top left of the screen.

How long he stays there depends on you and whether or not you press any of the cursor keys.

It's the endless REPEAT ... UNTIL loop formed by lines 90 and 170 that allows the keyboard to control the movement of the alien.

Each time round the loop the Electron prints the alien at a position on the screen given by *X* and *Y*. It then goes on to see if any of the four cursor keys have been pressed, and the alien is moved as necessary.

It does this testing by means of the INKEY function. INKEY can work in two ways. The first is when you follow it by a positive number inside brackets in a line like:

```
keypress=INKEY(200)
```

or

```
character=INKEY(100)
```

Here the INKEY holds up the program and waits for a key to be pressed. However it will only wait for a certain length of time which is specified by the number in brackets after the INKEY.

This figure tells the micro the number of hundredths of seconds that it is to hold up the program while it scans the keyboard.

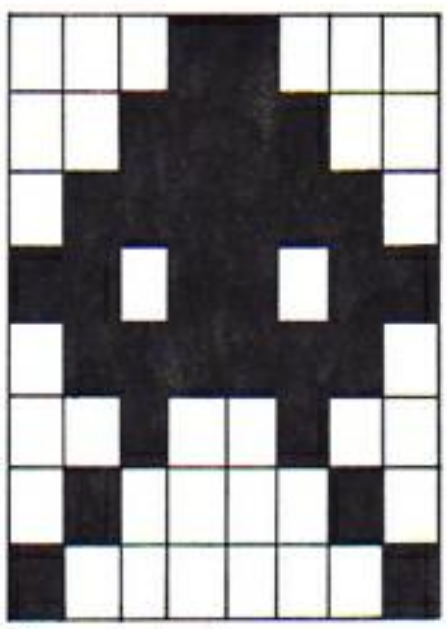


Figure I: An alien

When and if a key is pressed the Ascii value of that character is passed to the variable on the left of the equals sign.

If no key is pressed within the time limit the Electron just gives the variable a default value of –1 and gets on with the rest of the program.

In the first example we gave the Electron would wait for up to two seconds, scanning the keyboard to see if a key had been pressed.

Supposing that we had pressed the A key within the time limit, then the variable *keypress* would be given the value 65, the Ascii code for A.

If the time limit expired without a key being pressed then *keypress* would be given the value –1.

Having said all that, it's the second way of using INKEY that we're interested in. This is where the number in the brackets following the INKEY is negative.

Obviously this number can't refer to a time limit, unless the Electron can go backwards in time. What happens is that the negative number tells the Electron to go and see if a specific key is being pressed.

Each key on the keyboard has a particular negative number assigned to it, such as A is –66 and Z is –98. When the Electron reads a line like:

```
IF INKEY(-74) THEN PROCfire
```

it looks to see if the key specified by –74 (which is the Return key) is actually being pressed at that moment.

If it is, then the condition is TRUE and the program goes on to PROCfire. If the Return key isn't being pressed then the condition is FALSE and the program just goes onto the next line.

You'll notice that there isn't a time limit. If the key isn't pressed when it is tested then the program goes on to the next line straight away.

It also ignores any other keys that might be being pressed. It is only interested in the one key specified.

Page 159 of the Electron User Guide gives a full list of the negative numbers that can be used with INKEY and the keys that they refer to.

It's the use of INKEY with negative numbers that allows control of the aliens, as you'll see from lines 110 to 140. All four lines begin with an IF followed by a negative INKEY.

These lines test each of the four cursor keys in turn and, if

```
  10 REM GRAPHIC CONTROL
  20 REM BY IAN RODGERS
  30 REM use cursor keys
  40 MODE1
  50 VDU23,1,0;0;0;0;
  60 VDU23,224,24,60,126,2
19,126,36,66,129
  70 X=0
  80 Y=0
  90 REPEAT
 100 PRINTTAB(X,Y)CHR$224
 110 IF INKEY(-122) THEN X
=X+1:PRINTTAB(X-1,Y)" ":IF
X=39 THEN X=38
 120 IF INKEY(-26) THEN X=
X-1:PRINTTAB(X+1,Y)" ":IF X
=-1 THEN X=0
 130 IF INKEY(-42) THEN Y=
Y+1:PRINTTAB(X,Y-1)" ":IF Y
=31 THEN Y=30
 140 IF INKEY(-58) THEN Y=
Y-1:PRINTTAB(X,Y+1)" ":IF Y
=-1 THEN Y=0
 150 SOUND 1,-15,X,1
 160 SOUND 1,-15,32-Y,1
 170 UNTIL FALSE
```

Program I

Program ProObe

manoeuvres control

they are being pressed, they adjust the values of *X* and *Y* accordingly.

Line 110 tests to see if the right cursor is being pressed. If it is, it goes on to the rest of the line and adds one to the value of *X*.

For the moment we'll ignore the rest of the line and have a look at the following one.

Here the left cursor key is tested and, if it is being depressed, then one is subtracted from the value of *X*.

If you think about it you'll see that this means that when the loop gets round to actually printing the alien at the position *X, Y* the alien will have moved one place to the left or right, depending on the cursor key pressed (if any).

Similarly lines 130 and 140 test the up and down cursor keys and add or subtract one from the value of *Y* as necessary.

Each time round the loop the keys are tested in turn and the values of *X* and *Y* altered accordingly.

Since these variables determine the position of the alien at the beginning of each cycle through the loop you'll see that pressing the cursor keys has the effect of moving the alien.

Of course Ian didn't have to use the cursor keys. He could have used any of the keys on the keyboard.

However it's nice to have the arrows on the keys pointing in the direction that the alien will travel! Figure II shows what's happening.

Simple isn't it? But what about the parts of lines 110 to 140 that we ignored before?

There's nothing hard about them – they all work in the same way.

Let's take line 110 as an example. Here, as we already know, one is added to the value of *X* if the right cursor key is pressed.

This means that next time round the loop the alien will be printed at the same level but one space to the right.

However unless we do something about it the old alien will still be there. Obviously this isn't wanted, so the program prints a space over it.

The old position is found by subtracting one from *X* (*Y* will still have the same value).

If you can't follow that last part, remember that we've just added 1 to the value of *X* when the key was pressed, so we have to take it off again to find the old position.

All the final IF statement does is to make sure that the alien doesn't go off the edge of the screen. It does this by taking one off the value of *X* if it gets too close.

Lines 120, 130 and 140 work in exactly the same way, ensuring that if their particular key is pressed then the old alien is overwritten by a space. They also ensure that it doesn't go off the screen.

Lines 150 and 160 just produce the sounds. The pitch of the two notes is made dependent on *X* and *Y*. As *X* increases and the alien goes further right across the screen the note goes higher. Similarly when *Y* increases as the alien gets lower down the screen, the note decreases.

And that is that. As I said, it's a simple program, but it contains some useful concepts.

It shows how negative INKEYs can be used to test for specific keys being pressed without the program being delayed.

It also demonstrates how the values returned from those keys can be used both as the coordinates of a character and also to determine the pitch of two notes.

Finally, the program shows how to overprint with spaces, giving a simple animation effect.

Did I just say it was simple?

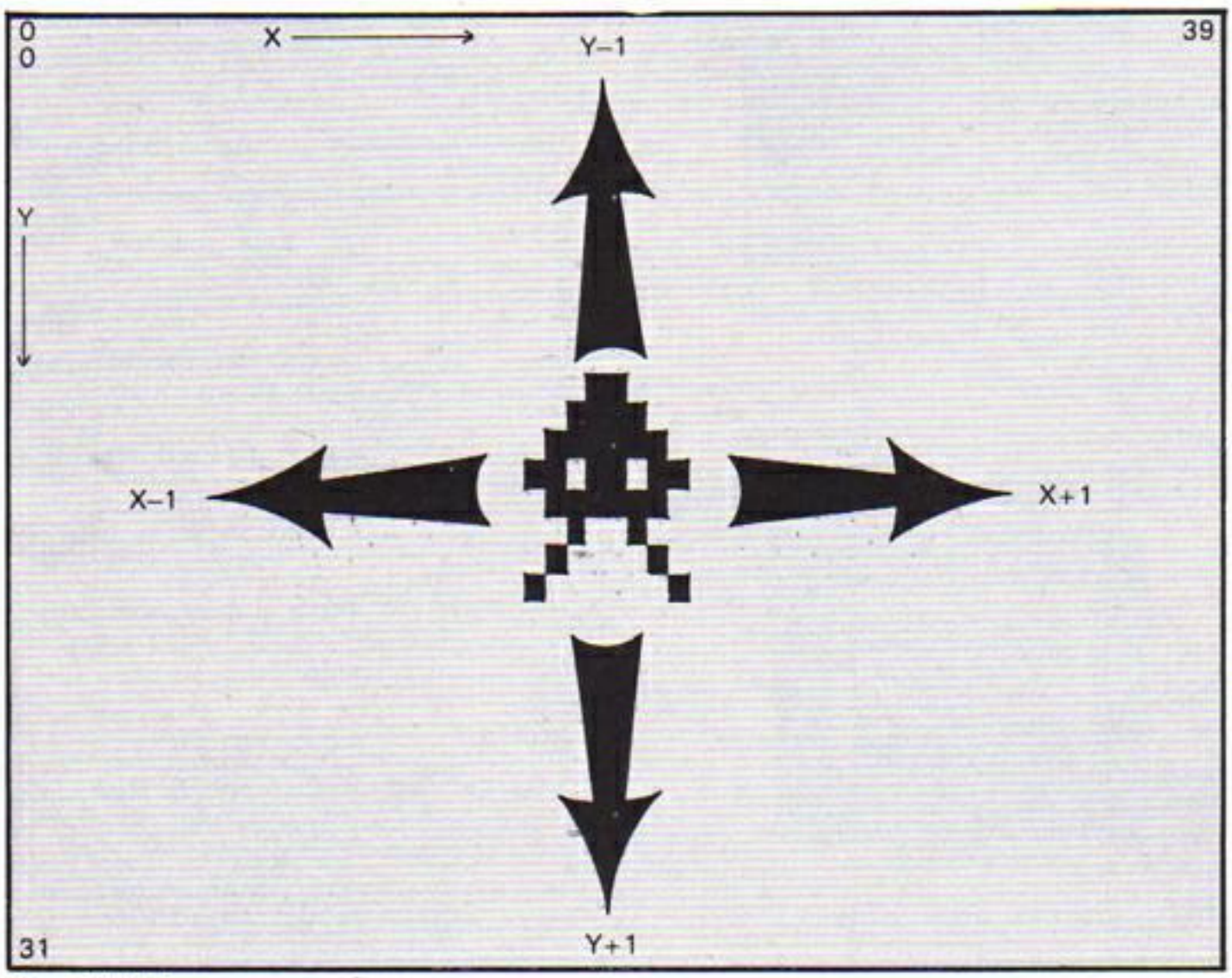


Figure II: Alien movement

DO YOUR INVADERS FLY?

OR DON'T THEY EVEN GET OFF THE GROUND?

Bring your designs
to life with

SIMONSOFT SPRITES VERSION TWO

FOR THE ELECTRON

SAVE £4
NOW ONLY £8.95
WHILE STOCKS LAST

Transfer your ideas for multi-coloured characters directly to the computer and screen with the easy to use grid-based generator program. As you design the sprites they are automatically stored in our sprite routine which lies hidden under your program. The machine code sprite routine will move the shapes at incredible speeds of fourteen (14) times that of ordinary basic – and the routine is controlled with simple commands from your own Basic program!

Simonsoft sprites are used in programs on the market now. We claim no royalties on programs using our sprite routines.

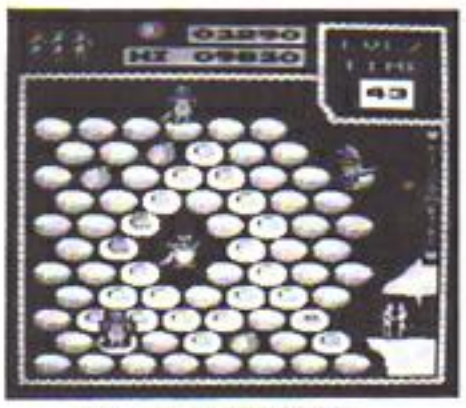
FEATURES:

- An ASTOUNDING FOURTEEN (14) FOLD INCREASE over the speed of ordinary basic
- GENERATOR PROGRAMS with which you design MULTI-COLOURED SPRITE CHARACTERS
- SUPERSPRITES of up to 24 × 24 pixels
- 48 SPRITES made up of 12 separate designs each with 3 clones
- INSTANT ANIMATION with two images per sprite that switch automatically
- ENLARGEMENT FACILITY OF ×2, ×3, ×4, ×5 sprite size
- BUILT IN COLLISION DETECTOR
- FLIGHT PATHS that sprites can follow automatically
- SPRITE LIBRARY of ready to use character designs
- 30 PROGRAMS – choose the routine with the features best suited to your own program. Hidden under your Basic program the routine need take as little as 1.5K memory.
- SAVE/LOAD your program and sprites AS A WHOLE
- COMPREHENSIVE COLOUR MANUAL
- INTRODUCTION PROGRAM and DEMONSTRATION GAMES

NEW RELEASE

OG THE CAVEMAN at £7.95 Electron

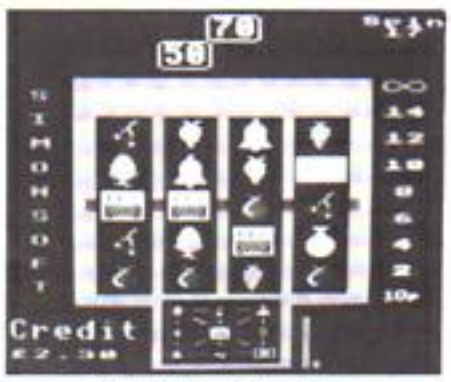
OG is in a calamity. His territory has been taken over by dinosaurs that lay eggs at a really hectic rate. Og must crush the brood before they hatch by jumping on to each and every egg. Og's problems are compounded by a pack of slavering dinosaurs hungry for his blood – and on higher levels by grand old Ma dinosaur herself. Og can use his magic staff to set a trap, but time is precious as Og knows all too well. Big bonuses can be scored for jumping on fruit, and an even bigger bonus for snatching a kiss from his waiting cavewife.



SCREEN PHOTOGRAPH

SUPERFRUIT at £5.95 for the Electron

Features full colour hi-res graphics, great sound effects, spinning reels, 5 × 4 display of fruit, "bounce" as each reel settles, nudge box, nudge gambles, two-way nudges, swap reels, hold reels, collect win, gamble any win, "loser's gamble" if first gamble is lost. Supergamble for the jackpot and a coin pile that shrinks and grows with your winnings. Separate instruction program. This implementation is in a class of its own.



SCREEN PHOTOGRAPH

A MUST for anyone who wants to see their Electron's graphics stretched to the very limit.

Please rush me _____ (Qty) SIMONSOFT SPRITES VERSION TWO for the Electron at **£8.95** each

_____ (Qty) OG THE CAVEMAN for the Electron at **£7.95** each

_____ (Qty) SUPERFRUIT for the Electron at **£5.95** each

Name _____

Address _____

_____

_____

Cut out this coupon and send it to:
SIMONSOFT, 25 Tatham Road, Abingdon, Oxon OX14 1QB
Telephone 0235 24140

PROGRAMMERS: we pay lump sums and/or royalties of up to 30% for EXCELLENT PROGRAMS

Casting Agency

Spaceships

From Steven Kelly, Liversedge, W. Yorkshire

```
VDU23,245,8,19,32,124,
111,35,16,8
VDU23,246,64,32,144,248,
216,16,32,64
VDU23,247,96,96,27,6,
7,27,97,96
VDU23,248,12,12,176,192,
192,176,12,12
```

Emu

From Robin Jenner, Edinburgh

```
VDU23,249,56,236,60,
8,24,16,48,118
VDU23,250,127,127,62,
28,8,8,8,60
```

Flowerpot

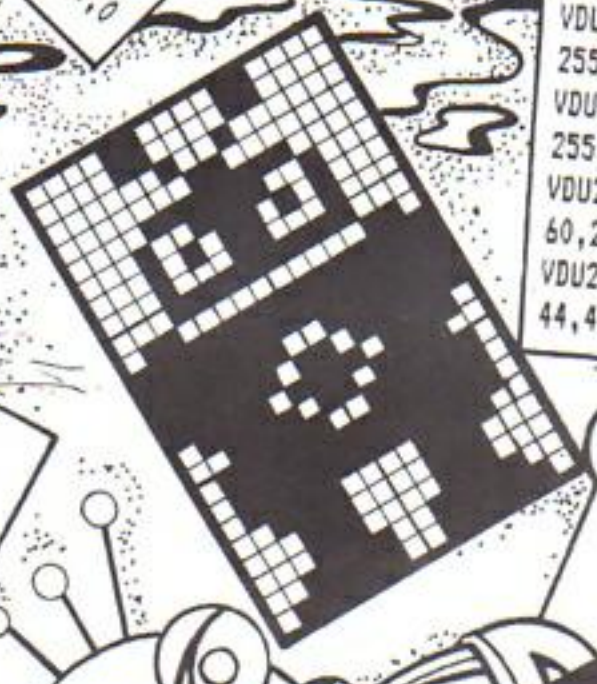
From Jason Owens, Batley, W. Yorkshire

```
VDU23,224,0,0,0,3,
4,4,131,225
VDU23,225,0,0,0,0,
128,136,30,60
VDU23,226,113,121,
61,23,1,1,1,7
VDU23,227,120,192,
16,120,222,12,8,224
VDU23,228,3,3,3,3,
3,1,0,0,
VDU23,229,192,192,
192,192,192,128,0,0
```

Electron Eddie

From Steven Jones, Ravenshead

```
VDU23,233,12,12,2,
1,15,19,21,17
VDU23,234,48,48,64,
128,240,200,168,136
VDU23,235,31,96,255,
255,254,253,123,61
VDU23,236,248,6,255,
255,127,191,222,188
VDU23,237,126,127,127,
60,20,20,62,126
VDU23,238,126,254,254,
44,40,40,124,126
```



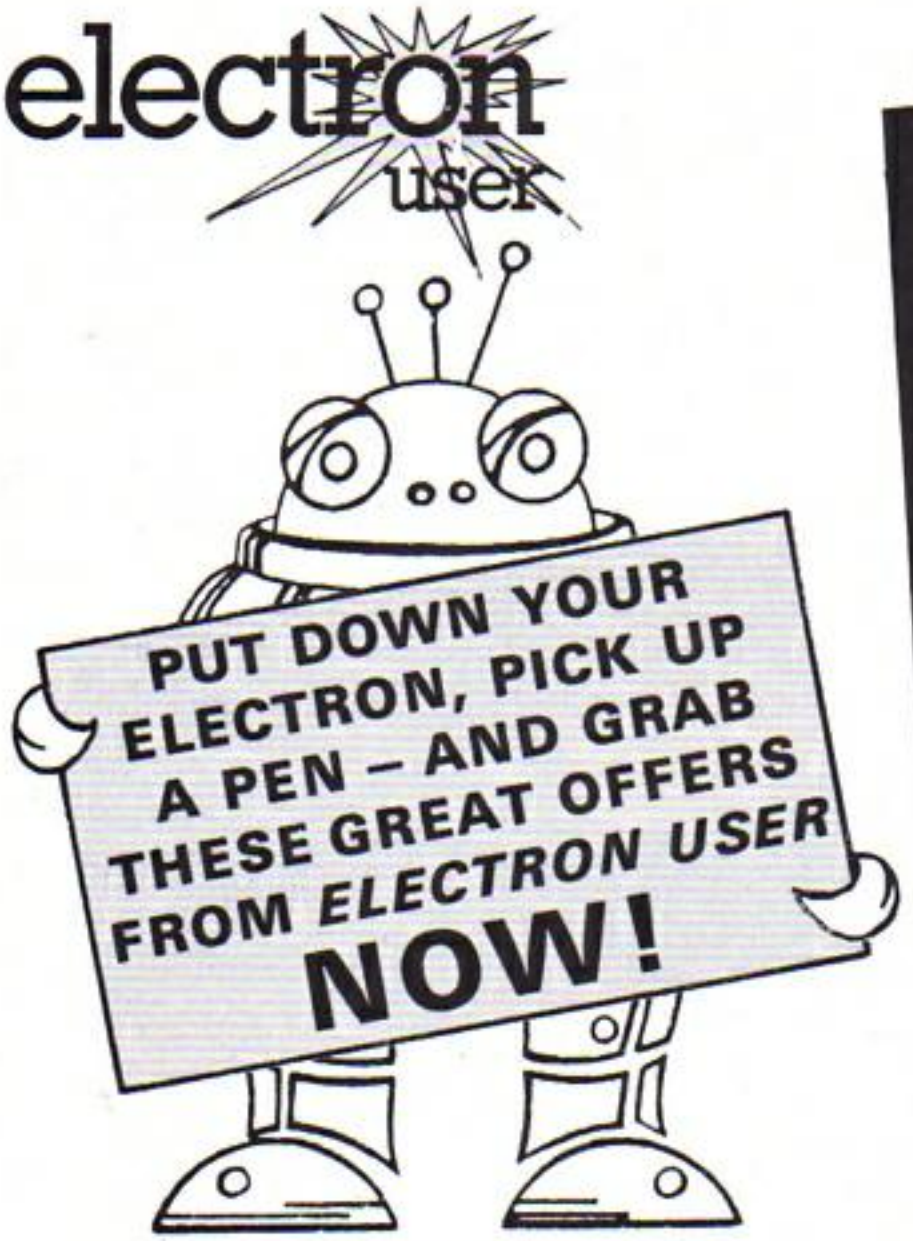
Squirrel

From A. and J. Rodgers, Wigan

```
VDU23,239,0,0,0,0,8,
60,104,60,28
VDU23,240,0,12,62,
126,124,120,120,120
VDU23,241,14,15,95,
127,15,7,7,7
VDU23,242,56,30,142,
198,198,230,230,246
VDU23,243,3,3,3,15,
120,0,0,0
VDU23,244,252,240,
224,192,0,0,0,0
```

HAVE you a favourite character you would like to see in this monthly feature in Electron User? Send your drawing of the character, together with the VDU23 statement, to: Shape Dictionary, Electron User, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

electron user



Be one of the first to get each issue

A subscription will ensure you get your own personal copy HOT OFF THE PRESSES month after month for the next year.

Every owner of an Electron – and everyone thinking of buying one – needs to get Electron User every month. It's the brightest, most authoritative yet completely independent guide to a machine that has so much potential you will never tire of reading about its remarkable capabilities.

You can buy Electron User from your local newsagent or station bookstall. Or you can take out a 12 months subscription and have it delivered to you by post.

Your Electron needs protecting!

Protect your Electron with our luxury dust cover made of soft pliable water-resistant vinyl, bound with strong cotton and decorated with *Electron User* logo.

£3.95

Keep your collection of *Electron User* complete with these handsome binders

Bound in attractive red pvc with the *Electron User* logo in gold blocking on the spine, this binder will hold 12 magazines firmly secured in place by metal rods. **£3.95**

FREE Cassette worth £3.75 if you subscribe NOW!

If you take out a subscription to Electron User now you will receive completely free one of the monthly cassettes of Electron User listings. Choose which one you want from those illustrated below.

This free gift is for a limited period, so subscribe now!

Cassette tapes of Electron User programs

Save typing in programs from *Electron User* by sending for these program-packed tapes.

£3.75 each

You can also take out a subscription for the 12 monthly tapes for £40.

ORDER FORM

All prices include postage, packing and VAT, and are valid to September 28.

Please enter number required in box £ p

Electron User annual subscription

UK £12 ☐
EIRE £13 (IR £16) ☐
Overseas (Surface) £20 ☐
Overseas (Airmail) £40 ☐

Selected free cassette _____ (month)
Commence with _____ issue TOTAL _____

Electron User introductory issues

£2.00 UK
£2.25 Overseas (Surface)

Complete set of 4 ☐

TOTAL _____

Electron User back issues

£1.25 UK
£1.50 Overseas (Surface)

February ☐
March ☐
April ☐
May ☐
June ☐
July ☐
August ☐

Airmail prices on application

TOTAL _____

Electron User tapes

£3.75
(UK & Overseas)

26 introductory programs ☐
Lunar Lander February ☐
Chicken March ☐
Spacehike April ☐
Rally Driver May ☐
Money Maze June ☐
Golf July ☐
Castles of Sand August ☐
Haunted House Sept ☐

TOTAL _____

Cassette tape annual subscription

£40 (UK & Overseas) ☐
Commence with _____ tape (state month) TOTAL _____

Dust Cover

£3.95
(UK & Overseas) ☐ TOTAL _____

Binder

£3.95 UK
£5.00 Overseas ☐ TOTAL _____

Payment: please indicate method (✓) **TOTAL** _____

☐ Access/Mastercharge/Eurocard
☐ Barclaycard/Visa
☐ American Express

Card No. _____

Expiry Date _____

☐ Cheque/PO made payable to Database Publications Ltd

Name _____

Address _____

_____

Signed _____

Send to: *Electron User*, FREEPOST, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

(No stamp needed if posted in UK) *Please allows 28 days for delivery*

You can also order by phone

Telephone: 061-480 0171
24 hours

Don't forget to quote your credit card number and give your full address

TREVOR ROBERTS assesses the official hardware extension for your micro

The Electron and the Plus 1 —made for each other

THE first thing that strikes you about the Plus 1, the official hardware extension for the Electron, is that it looks neat. It fits on the back of the Electron, screwing firmly into place and immediately blends in with it, seeming to have always been there.

As a colleague and non-Electron fan said when he saw the combination, "That makes the Electron look like a proper micro".

The guy's an ignorant yahoo, but I can see what he means. And it's not just the appearance of the Electron that the Plus 1 improves, it also expands its powers enormously.

It allows the Electron to use a printer, joysticks and (when there are some) ROM file cartridges.

To be more accurate, it's not just a joystick capacity that the Plus 1 gives the Electron, it gives it an analogue interface.

This analogue port allows the Electron to measure smoothly varying electric currents and convert them to a numeric form that the Electron can handle.

What this means in practice is that all sorts of electrical devices can be attached to it, from heat and light sensors to heart rate monitors. Joysticks are, however, the most obvious use of the port.

As it is an analogue port, it's no surprise that the joysticks that are used with the Plus 1 are analogue joysticks. Most of the previous interfaces for the Electron have used the switched, Atari-style joysticks.

This leads to a problem in that, while the Plus 1 will allow the use of joysticks with Acornsoft games, games with a joystick option from other software manufacturers may not work.

No doubt in time software houses will incorporate the Plus 1 joystick handling routines in their software. Until then it's a serious mark against the official add-on.

Having said that I got hold of a set of Voltmace analogue joysticks (excellent hardware) and used them with Acornsoft's Monsters and Starship Command.

Once I'd figured out how the joystick movements replaced the more familiar keys the difference was remarkable. Using joysticks really improves games.

The second major feature of the Plus 1 is its Centronics-compatible parallel interface which allows the Electron to use a wide range of printers.

Working on *Micro User* and *Electron User* has meant that I've had a lot of experience using the BBC Micro's printer facility, which is excellent.

The Plus 1 gives the Electron exactly the same capability. I attached the printer to my Plus 1 and the rest was plain sailing.

Normal print, italics, bold print, I could get them all, simply and easily. When I say it was just like using the BBC, you can take that as praise indeed.

The Plus 1 manual explains all the previously unlisted *FX calls that you need to know. The manual is clear, concise and useful, but newcomers to using a printer should be warned that the manuals that come with printers are usually pretty awful.

Any difficulties you may have using your Plus 1 for printing will almost certainly come from that area and not the hardware itself.

As to the Plus 1's ROM cartridge facility there's not a lot I can say. Despite the leaflet's promise that some were available, none came with the Plus 1 and Acorn's public relations firm couldn't supply me with any to test out. However, if the manual is to be believed, they sound quite promising.

The cartridges work in two ways. The first uses them as a sort of very fast read-only cassette system for entering games and applications software. This promises to be far quicker than the usual cassette filing system.

The second method is for language cartridges such as Lisp.

Here the language itself is used as an alternative to the Basic already in the Electron, leaving you the normal amount of memory space for programs in the new language. I look forward to that.

The Plus 1 has room for two of these cartridges at once, and when they arrive, they will significantly increase the Electron's potential as a serious micro.

Also, if hints that an RS423 interface is being developed for the cartridge slots have a basis in fact, then the Plus 1 will allow the Electron to enter the growing world of computer communications.

However that's for the future. What Plus 1 owners get for their money now is the capability to use joystick (with Acornsoft games), hang peripherals on the analogue port, and use parallel printers.

There's also the potential for using cartridge software when it becomes available.

By providing the analogue port and cartridge slots it opens up the Electron to the outside world, and I doubt if it will be long before enthusiasts and manufacturers take advantage of this.

All in all, it's a very good piece of hardware that takes the Electron into the same league as other, more expensive micros.

While the lack of joystick-compatible software and the cartridges is a mark against it, I have little doubt that time will remedy this.

As it is, the Plus 1 has to be the most useful Electron expansion unit to come on the market.

LE BOX

Electron Disc Interface

Another first from Pace . . .

For the first time ever, it is now possible to use disc drives with your Acorn Electron computer.

With Le Box, a revolutionary new accessory for the Electron, it is possible to use BBC type discs and 8K sideways ROMs on the BBC's little brother.

Now you can create true random access files on the Electron and greatly reduce the Loading, Saving and Access times for handling programs and data – a tremendous increase in the Electron's potential.

The interface is supplied in a self-contained unit matched to the Electron colour, complete with its own power supply and one of the superb range of Pace 5.25" disc drives.

All the following commands provided by the advanced Amcom disc filing system are also available on the new Electron interface:

*ACCESS	Allows files to be 'locked' thus eliminating the possibility of accidental deletion.
*ADD	Allows new lines to be added to files previously created with the *BUILD command.
*BACKUP	Copies a complete disc.
*BUILD	Allows the creation of text files directly from the keyboard.
*CLEAR	Clears the catalogue of the current disc.
*COMPACT	Moves all files to the beginning of a disc leaving all free space at the end.
*COPY	Copies individual files or groups of files from one disc to another.
*DELETE	Deletes individual files or groups of files.
*DIR	Sets the current directory.
*DRIVE	Selects the current drive number.
*DUMP	Gives a HEX and ASCII dump of a file on the screen.
*ENABLE	Allows 'dangerous' commands to be used.
*FORMAT	Formats a disc.
*INFO	Displays catalogue information about files.
*LIB	Sets the current library.
*LIST	List programs from disc to the screen.
*MOVE	Selectively copies files.
*OPT2,n	Sets the number of sectors per track to n.
*OPT3,n	Sets the number of tracks per disc to n.
*OPT5,n	Sets the start of the disc buffer (see OPT7).
*OPT6,n	Determines the amount of file information displayed.
*OPT7,n	Sets the length of the disc buffer.
*OPT8,n	Verifies track n.
*RENAME	Allows files to be re-named.
*SPACE	Displays the total amount of free space remaining on a disc.
*TITLE	Allows a disc to be titled.
*TYPE	Lists an ASCII file from disc to the screen.
*WIPE	Allows selective deletion of files.

In addition to the commands listed above you may also use the MOS commands LOAD, *LOAD, SAVE and *SAVE for loading and saving either Basic programs or blocks of machine code.

Also included on the board are sockets to accept 8k sidways ROMs, yet another superb feature normally missing on the Electron. Sideways ROM's allow instant access to the powerful software packages which remain permanently resident in your machine.

Le Box includes all cabling for simple connection to the Electron's rear edge connector along with a comprehensive, easy to follow manual. The units also include externally switchable drive select lines to enable copying to and from an external disc drive. Auxilliary data and power sockets allow the simple connection of other drives or accessories.

The unit is normally supplied with a single sided 40 track drive giving 100K of storage capacity per disc. Other drives may be fitted on request including 40/80 switchable units offering 400K capacity. Please ring for details.

PACE SOFTWARE LTD.
92 NEW CROSS STREET,
BRADFORD BD5 8DS.
Tel: (0274) 729306 Telex: 51564.

After FOUR user-packed shows we are moving back to the North!

Get ready for the

UMIST, Manchester, Friday August 31,

OFFERS GALORE

Everything on display at the show will be for sale – often at special low show prices. So you'll easily be able to save much more than your admission ticket in your first five minutes at the show!

Hundreds of new games, educational and business programs, plus all the latest hardware add-ons for the BBC Micro and Electron – and all for you to try and buy.

This is where it all began!

It was here, in June, 1983, that we held our first show, and it attracted 12,500 enthusiastic BBC Micro fans. The UMIST building is only five minutes from British Rail's Piccadilly main line station – and there is plenty of parking space available on the campus. In fact, it's an ideal conference location.

EXHIBITORS INCLUDE

Acorn Computers, Advanced Memory Systems, Aries Computers, Bucon, Cambridge Microprocessor Services, Clares Micro Supplies, Clwyd Technics, Computer Concepts, Com-soft, Cumana, Dynabyte Software, Enterprise Technology Computing, Feedback Instruments, Golem, HCR Electronic Sales, Holiday Brothers, Kansas City Systems, M.P. Software and Services, MRM Software, M.T. Direct, Mr Floppy, Manchester University Press, Micro-Aid, Micro-Aids, Micropower, Micro Simplex, Microware, Microvitec, NSC Computer Shops, National Micro Centres, News-soft, Opus Supplies, Pace Software Supplies, Specialised Education Software and Services, System, Universal Communications, Video Electronics, Viglen Computers, Wise-Owl Publications, Wizard Development.

FIFTH great...

to Sunday September 2

ELECTRON & BBC MICRO USER SHOW

You'll find all the answers to your problems at our

Walk-in Forum

We are bringing back the much talked-about feature of our first Manchester show.

Throughout the weekend some of Britain's top experts on the BBC Micro and Electron will be making guest appearances at the forum, held in a lecture theatre adjoining the two exhibition halls.

Whether you're a beginner or an experienced user, they will be delighted to answer all your questions.

Reduced prices for School/College Groups

Entry only £1 per student if bookings are made in advance. Send your cheque (made payable to Database Publications) and SAE to:

Electron & BBC Micro User Show
68 Chester Road, Hazel Grove
Stockport SK7 5NY Tel: 061-456 8383

Valid for a minimum of 10 people

SAVE MONEY with our Special Travel Offer

Visitors to the Show can obtain cut-price rail tickets from ANY station in the United Kingdom

For full details write to:

Travel Offer, P.O. Box 1, St. Albans AL1 4ED with SAE or Telephone: St. Albans 34475 quoting: The Electron & BBC Micro User Show.

On sale will be

BBC Micros, Electrons, Teletext adapters, Torch disc packs, BBC Buggies, Second Processors, ROM Expansion Boards, Grafpads, Books, Joysticks, Interfaces, Disc drives, Data recorders, Lightpens, Modems, Speech Synthesisers, Carrying Cases, Cables, Digitizers, VDU stands, Graphics tablets, ROM chips, Monitors, Printers . . .

PLUS *many thousands of software programs – games, educational and business packages.*

This voucher is worth 50p per head

ELECTRON & BBC MICRO USER SHOW

By handing in this voucher at the door you save 50p off the normal admission price of £2 (adults) and £1.50 (children).

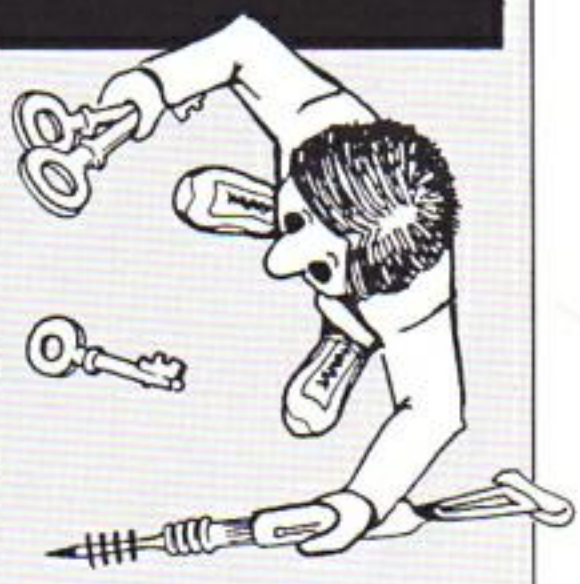
(Valid for a maximum of 4 people)

10am - 6pm, Friday, 31 August
10am - 6pm, Saturday, 1 September
10am - 4pm, Sunday, 2 September

Renold Building, UMIST, Manchester.

Number attending: 1 2 3 4

Haunted House listing



From Page 33

```
   06GOTO17
   1dx%=-1:dy%=0:d%=0:RETU
RN
   2dx%=1:dy%=0:d%=1:RETUR
N
   3dx%=0:dy%=1:d%=2:RETUR
N
   4dx%=0:dy%=-1:d%=3:RETU
RN
   5dx%=0:dy%=0:RETURN
   6sc%=sc%+75:FORloop%=10
0TO200STEP10:SOUND1,1,loop%
,1:NEXT:PROCscore:RETURN
   7sc%=sc%+95:SOUND1,2,10
0,3:PROCscore:RETURN
   8VDU17,3,31,1,doory%,24
2,10,8,242,10,8,242:dx%=0:d
y%=0:IFmekey%>4ANDhit%=need
% end%=TRUE:RETURN ELSE RET
URN
   9PRINTTAB(x%,y%)bl$;:scr
1%=FNscrn(x%+dx%,y%+dy%):s
cr2%=FNscrn(x%+dx%,y%+dy%+1
):x%=x%+dx%:y%=y%+dy%:dx%=0
:dy%=0:PROCdead:RETURN
   10mekey%=mekey%+1:FORloo
p%=190TO250STEP10:SOUND1,-1
,loop%,1:NEXT:sc%=sc%+15*me
key%:PRINTTAB(7,2)mekey%;:P
ROCscore:RETURN
   11RETURN
   12*FX210,1
   13RETURN
   14REPEAT:UNTILGET=32:RET
URN
   15*FX210,0
   16RETURN
   17MODE5:ON ERROR GOTO 26
1
   18:
   19REM Initialise and ass
emble M/C
   20*FX11,0
   21PROCinit
   22PROCassemble
   23:
   24REM Main Game loop
   25:
   26REPEAT
   27A%=135:sc%=0:li%=3:lev
%=1
   28PROCinstruct
   29PROCdrawscreen:PROCgam
e
   30IFli%>0 PROCnewscreen:
GOTO29
   31PROCalldead
   32UNTIL FALSE
   33END
   34REM Read the screen fu
nction
   35DEFFNscrn(q%,w%):VDU31
,q%,w%:=(USR(&FFF4)AND&FF00
)DIV&100
   36REM Pause a while
   37DEF PROCw(pause%):FORl
oop1%=1TOpause%:NEXT:ENDPRO
C
   38REM Print the score
   39DEF PROCscore:VDU17,3,
31,4,1:PRINTLEFT$("00000",5
-LEN(STR$sc%))sc%;:ENDPROC
   40REM Print the lives
   41DEF PROClives:li%=li%-
(li%=-1):VDU17,2:man$=man$(
3,1)+CHR$11:bla$=bl$+CHR$11
:PRINTTAB(10,1)STRING$(li%,
man$)STRING$(4-li%,bla$);
   42ENDPROC
   43REM print the level
   44DEF PROClevel:COLOUR1:
PRINTTAB(6,31)"L=";:COLOUR3
:PRINTLEFT$("00",2-LEN(STR$
lev%))lev%;:ENDPROC
   45REM Play a tune
   46DEF PROCtune(string$,c
hannel%,duration%):FORloop%
=1TOLEN(string$):note%=(ASC
(MID$(string$,loop%,1))-43)
*5
   47SOUND1,channel%,note%,
duration%:NEXT:ENDPROC
   48REM Move the man
   49DEF PROCman:in%=INKEY0
:IFin%>0GOSUBkey%(in%)
   50scr1%=FNscrn(x%+dx%,y%
+dy%):scr2%=FNscrn(x%+dx%,y
%+dy%+1):IFscr1%<>32GOSUBsc
reen%(scr1%)
   51IFscr2%<>32GOSUBscreen
%(scr2%)
   52PRINTTAB(x%,y%)bl$TAB(
x%+dx%,y%+dy%)man$(d%,man%)
;:x%=x%+dx%:y%=y%+dy%:man%=
man%+1:IFman%=3man%=1
   53IFINKEY-1PROCfire
   54ENDPROC
   55REM Play the game
   56DEF PROCgame:end%=0:RE
PEAT:PROCman:CALLCODE:?rnd%
=RND(4):?(di%+RND(15))=RND(
4):IF?mhit%=1PROCdead
   57PROCman:bo%=bo%-inc%:I
Fbo%<0PROCtimeup ELSEPRINTT
AB(14,2)LEFT$("00000",5-LEN
(STR$bo%))bo%
   58UNTILend%:ENDPROC
   59REM You are dead
   60DEF PROCdead
   61li%=li%-1:PROClives:IF
li%=0end%=TRUE
   62FORloop%=1TO10:FORloop
2%=3TO1 STEP-1:VDU17,loop2%
:SOUND1,1,loop2%*85-loop%*1
0,1:PROCw(40)
   63PRINTTAB(x%,y%)man$(3,
1);:NEXT:PROCw(10):NEXT:VDU
17,3,31,x%,y%,237,10,8,238:
FORloop%=50TO0STEP-3:SOUND1
,-1,loop%,1:NEXT:SOUND&0000
,-1,4,10:PROCw(200)
   64IFend%ENDPROC
   65*FX15,0
   66PROCtune("101010167676
76710101078787878101010109;9;
9;9;9;8765432101010101",-1,
1)
   67FORloop%=1TO?number%:I
Floop%?xp%=x% IFloop%?yp%=y
%ORloop%?yp%=y%+1 loop%?dea
d%=1:hit%=hit%+1
   68NEXT
   69PRINTTAB(x%,y%)man$(d%
,man%);:PROCstartgame
   70?mhit%=0:PRINTTAB(x%,y
%)bl$;:IFbo%<2000bo%=2000
   71ENDPROC
   72REM Youv'e run out of
time
   73DEF PROCtimeup:PRINTTA
B(14,2)"00000":VDU17,2:FORl
oop%=1TO20:PRINTTAB(14,1)"B
ONUS";:SOUND1,-1,255-(loop%
*12),1:PROCw(150):PRINTTAB(
14,1)"      ";:PROCw(160):NE
XT
   74PRINTTAB(14,1)"BONUS";
:VDU17,3:bo%=5000:PRINTTAB(
14,2)LEFT$("00000",5-LEN(ST
R$bo%))bo%;
   75PROCdead
   76ENDPROC
   77REM Fire your laser
   78DEF PROCfire:h%=0:IFd%
>1ENDPROC ELSESOUND1,1,150,
5:GCOL3,2:IFd%=0PROCleftfir
e ELSEPROCrightfire
   79*FX20,1
   80loop%=0:REPEAT:IF?(yp%
+loop%)=y%+1AND?(dead%+loop
%)=0THENIFx%>?(xp%+loop%)AN
Dd%=0ORx%<?(xp%+loop%)ANDd%
=1 PROChit(loop%)
   81loop%=loop%+1:UNTILh%=
1ORloop%=need%+1:IFh%=0ANDd
%=0PROCleftfire ELSEIFh%=0A
NDd%=1PROCrightfire
   82ENDPROC
   83REM Draw the laser fir
e
   84DEF PROCleftfire:MOVE6
3,980-(y%*32):PLOT21,x%*64,
980-(y%*32):ENDPROC
   85DEF PROCrightfire:MOVE
x%*64+64,980-(y%*32):PLOT21
,1215,980-(y%*32):ENDPROC
   86REM Youv'e hit an alie
n
   87DEF PROChit(num%):SOUN
D0,-1,4,2:hit%=hit%+1:h%=1:
IFd%=0PROCleftfire ELSEPROC
rightfire
   88xpos%=?(xp%+num%):ypos
%=?(yp%+num%):?(dead%+num%)
=1:FORloop%=1TO4:FORloop2%=
1TO3:VDU17,loop2%,31,xpos%,
ypos%,246:SOUND1,2,loop%*7+
loop2%,1:NEXT,
   89VDU31,xpos%,ypos%,32:s
c%=sc%+55:PROCscore:ENDPROC
   90REM Completed the scre
en
   91DEF PROCnewscreen:FORl
oop%=0TO255STEP9:SOUND1,1,l
oop%,1:PROCw(20):SOUND1,2,2
55-loop%,1:PROCw(20):NEXT
   92sc%=sc%+bo%:COLOUR129:
PROCscore:SOUND&0011,3,150,
2:PROCw(3000):COLOUR128:lev
%=lev%+1
   93IFlev%=5li%=li%+1:PROC
lives:PROCtune("01234567890
12345678922334455667788 99;;
::::",3,2):PROCw(900):SOUND
&0011,0,0,0
   94PRINTTAB(14,2)"00000":
PROCw(200)
```

→

Haunted House listing

From Page 53

```
  95PRINTTAB(3,20)"HAUNTED
 HOUSE!"
  96SOUND1,2,20,10:GCOL3,1
29:CLG:SOUND1,2,50,10:GCOL3
,130:CLG:SOUND1,2,100,10:GC
OL3,131:CLG:SOUND1,2,150,10
:SOUND1,2,200,10:CLS
  97ENDPROC
  98REM Draw the screen
  99DEF PROCdrawscreen:CLS
:RESTORE127
 100FORloop%=1TO3:VDU19,lo
op%,0;0;:NEXT:MOVE0,447
 101GCOL0,1:FORloop%=1TO33
:READx1%,y1%:DRAWx1%,y1%:NE
XT
 102MOVE0,447
 103FORloop%=1TO32:READx1%
,y1%,x2%,y2%:DRAWx1%,y1%:DR
AWx2%,y2%:DRAWx1%,y1%:NEXT
 104MOVE223,1007:PLOT21,12
47,1007:PLOT21,1247,911:PLO
T21,223,911:PLOT21,223,1007
:PROCscore:PROClives:PROCle
vel
 105bo%=4500+lev%*500:inc%
=15+(lev%*5):IFinc%>150inc%
=150
 106VDU17,1,31,14,1:PRINT"
BONUS":VDU17,3,31,14,2:PRIN
TLEFT$("00000",5-LEN(STR$bo
%))bo%;
 107VDU17,1,31,5,2,244,17,
2,61,17,3,48
 108key%=0:d%=2:dx%=1:dy%=
0:man%=1:x%=2:y%=9:PRINTTAB
(x%,y%)man$(d%,man%);
 109RANDOMISE=RND(-TIME)
 110PROCobjects(RND(4),240
,4)
 111PROCobjects(5,244,3)
 112IFlev%DIV2=lev%/2 PROC
objects(RND(3),243,1)
 113IFlev%>4 PROCobjects(R
ND(4),239,3) ELSEPROCobject
s(RND(3),247,1)
 114?number%=4+lev%:IF?num
ber%>12?number%=12
 115PROCaliens
 116doory%=8+RND(2):VDU17,
3,31,1,doory%,242,10,8,242,
10,8,242,17,3
 117VDU19,1,1;0;19,2,6;0;1
9,3,3;0;:mekey%=0:hit%=0:ne
ed%=?number%:PROCstartgame
 118end%=0
 119ENDPROC
 120DEF PROCstartgame
 121*FX20,1
 122KEY$=GET$:IFKEY$="Z"d%
=0ELSEIFKEY$="X"d%=1ELSEIFK
EY$="/"d%=2ELSEIFKEY$=":"d%
=3ELSE 122
 123dx%=dx%(d%):dy%=dy%(d%
):*FX15,0
 124scr1%=FNscrn(x%+dx%,y%
+dy%):scr2%=FNscrn(x%+dx%,y
%+dy%+1):IFscr1%<>326OSUBsc
reen%(scr1%)
 125IFscr2%<>326OSUBscreen
%(scr2%)
 126ENDPROC
 127DATA0,447,15,517,47,51
1,47,771,0,831,63,831,63,89
5,127,959,319,863,511,863,7
03,895,895,863,1087,895,108
7,831,1215,831,1279,767,121
5,671,1279,639,1279,319,121
5,255
 128DATA1279,159,1151,31,9
59,0,895,31,831,0,703,63,38
3,63,255,0,127,63,63,159,0,
255,63,383,0,447
 129DATA63,447,0,447,127,5
11,63,511,127,751,41,779,63
,799,0,831,127,799,63,831,1
27,831,63,895,143,879,127,9
59,351,767,319,863,511,767,
511,863,703,799,703,895,863
,767,895,863,1055,815,1087,
895,1087,735,1087,831,1183,
735,1215,831,1215,735
 130DATA1279,767,1151,671,
1215,671,1183,607,1279,639,
1183,351,1279,319,1151,255,
1215,255,1207,163,1279,159,
1087,63,1151,31,959,31,959,
0,895,63,895,31,831,31,831,
0,671,159,703,63,415,159,38
3,63,271,63,255,0,159,111,1
27,63,95,175
 131DATA63,159,63,271,0,25
5,127,399,63,383,63,447,0,4
47
 132REM run out of lives
 133DEF PROCalldead:VDU17,
3,31,x%,y%,237,10,8,238
 134GCOL0,0:VDU17,2:PRINTT
AB(5,18)"GAME  OVER";:PROCw
(4000):FORloop%=0TO639STEP8
:MOVEloop%,0:DRAWloop%,1023
:MOVE1279-loop%,0:DRAW1279-
loop%,1023:SOUND&0011,2,loo
p%DIV3,2:NEXT
 135GCOL0,129:CLG:GCOL0,12
8:CLG
 136ul$=STRING$(20,"_"):VD
U17,1,31,0,8:PRINTul$;:VDU1
7,3,31,0,10:PRINT"You score
d ";sc%:PRINT''"  on level
";lev%
 137VDU17,1,10,10:PRINTul$
;:VDU17,2,10:IFsc%>hi%(10)P
RINT"Your'e on the high  s
core table.";:hi%(10)=sc%:h
i$(10)="":FORloop%=100TO255
STEP10:SOUND1,1,loop%,1:NEX
T
 138VDU17,1,13,10:PRINTul$
;:VDU17,3
 139A=INKEY(200):FORloop%=
9TO1STEP-1
 140IFhi%(loop%)<hi%(loop%
+1)PROCswap
 141NEXT:GCOL3,129:CLG:GCO
L0,128:CLG:title$="Today's
High Scores "
 142VDU17,1,31,0,0:PRINTul
$;:VDU17,2:PRINTTAB(0,2)tit
le$;:VDU17,1:PRINTul$;:VDU1
7,3
 143hi%=0:row%=0:FORloop%=
1TO10:col%=6+loop%*2
 144VDU17,3:PRINTTAB(0,col
%)LEFT$("00000",5-LEN(STR$h
i%(loop%)))hi%(loop%);:VDU1
7,1,31,5,col%,ASC("-"),17,2
,31,6,col%:PRINThi$(loop%);
:IFhi$(loop%)=""row%=col%:h
i%=loop%
 145NEXT:IFhi%>0PROCinputn
ame(hi%)
 146VDU17,1,31,0,30:PRINTu
l$;:VDU31,0,27:PRINTul$;:VD
U17,2,31,0,29:PRINT"Press a
 key to start";:VDU17,3
 147*FX15,0
 148key$=INKEY$10:IFkey$="
"title$=RIGHT$(title$,1)+LE
FT$(title$,19):COLOUR3:PRIN
TTAB(0,2)title$;:GOTO148
 149ENDPROC
 150DEF PROCinputname(num%
):VDU23,1,1,0;0;0;0;0;:*FX1
5,0
 151VDU17,3:INPUTTAB(6,row
%)hi$(num%):VDU17,2:hi$(num
%)=LEFT$(hi$(num%),14):PRIN
TTAB(0,row%+1)STRING$(20,"
")TAB(6,row%)hi$(num%)
 152VDU23,1,0,0;0;0;0;0;:S
OUND1,2,50,2:PROCw(30):ENDP
ROC
 153ENDPROC
 154DEF PROCswap:hi%=hi%(l
oop%):hi%(loop%)=hi%(loop%+
1):hi%(loop%+1)=hi%:hi$=hi$
(loop%):hi$(loop%)=hi$(loop
%+1):hi$(loop%+1)=hi$:ENDPR
OC
 155REM Instructions
 156DEF PROCinstruct
 157VDU22,4:VDU19,0,4;0;19
,1,6;0;:VDU23,1,0,0;0;0;0;0
;
 158VDU17,0,17,129:title$=
"     HAUNTED HOUSE  by Pet
er Scott.     ":PRINTTAB(0,
1)title$;:VDU17,1,17,128
 159PRINT''"You control a s
mall man who is trapped  in
 a haunted house.In every r
oom,there  are various obst
acles which you must  avo
id and also various goodies
 you can  collect for bonus
 boints."
 160PRINT"  To get out of
the room,you collect allthe
 keys and shoot all the ali
ens with  your laser.The it
ems you get points for coll
ecting are swords,diamonds
and the  keys,but you mustn
't hit the red cans,  the a
liens or the ghosts."
 161PRINTSTRING$(40,"_");:
VDU17,0,17,129:PRINT"  USE
THESE KEYS TO CONTROL THE M
AN :-  ":VDU17,1,17,128
 162PRINT'"'Z'=left   'X'=
right   '?'=down        '*
'=up     'Q'=quiet   'S'=so
und on     'P'=pause,SPACE
starts 'SHIFT'=fire,if  'ES
CAPE'=restart game  facing
 sideways"
```

Haunted House listing

From Page 54

```
163PRINT''"After you colle
ct everything,make your wa
y to the door,in the top le
ft of the screen,and you w
ill exit the room,and mov
e onto the next,harder scre
en.You getan extra life if
you pass room 4."
164VDU17,0,17,129,31,0,30
:PRINT"    Press any key to
 start the game!    ";:*FX1
5,0
165key$=INKEY$3:IFkey$=""
title$=RIGHT$(title$,1)+LEF
T$(title$,39):VDU17,129,17,
0,31,0,1:PRINTtitle$;:GOTO1
65
166PRINTTAB(0,1)STRING$(4
0," "):PROCtune("7642764275
7;741;741;7417484;742222",-
1,2):VDU22,5:VDU19,1,1;0;19
,2,6;0;19,3,3;0;23,1,0,0;0;
0;0;0
167ENDPROC
168REM Initialise graphic
s etc.
169DEF PROCinit:RESTORE17
6
170@%=0:*FX4,1
171VDU19,0,0;0;19,1,1;0;1
9,2,6;0;19,3,3;0;23,1,0,0;0
;0;0;0;
172FORloop%=237 TO 255
173READa%,b%,c%,d%,e%,f%,
g%,h%
174VDU23,loop%,a%,b%,c%,d
%,e%,f%,g%,h%
175NEXT loop%
176DATA56,124,84,84,124,4
0,40,56
177DATA146,198,108,48,24,
108,198,130
178DATA129,90,60,126,94,4
4,90,129
179DATA17,27,14,12,28,54,
98,192
180DATA0,0,0,170,85,0,0,0
181DATA170,85,170,85,170,
85,170,85
182DATA60,66,129,195,255,
255,126,60
183DATA0,64,160,191,165,6
9,0,0
184DATA255,255,255,255,25
5,255,255,255
185DATA60,90,165,165,153,
255,189,126
186DATA60,126,255,153,187
,255,255,165
187DATA255,60,60,60,90,10
2,102,195
188DATA24,60,126,60,189,2
4,189,255
189DATA102,97,113,28,9,11
9,119,64
190DATA102,97,113,62,28,1
06,118,71
191DATA28,62,120,100,67,3
8,56,44
192DATA102,134,142,124,14
4,238,238,2
193DATA102,134,142,124,56
,86,110,226
194DATA56,124,30,38,194,1
00,56,52
195DIMkey%(139),screen%(1
59),man$(4,2),dx%(3),dy%(3)
,hi$(10),hi%(10)
196FORloop%=0TO3:READdx%(
loop%),dy%(loop%):NEXT
197DATA-1,0,1,0,0,1,0,-1
198FORloop%=0TO139:key%(l
oop%)=11:NEXT
199FORloop%=1TO11:READnum
b%,value%:key%(numb%)=value
%:NEXT
200DATA90,1,122,1,88,2,12
0,2,63,3,47,3,42,4,58,4,81,
12,80,14,83,15
201FORloop%=0TO159:screen
%(loop%)=11:NEXT
202FORloop%=1TO8:READnumb
%,value%:screen%(numb%)=val
ue%:NEXT
203DATA0,5,143,6,144,7,14
6,8,147,9,148,10,150,9,151,
9
204FORloop%=1TO10:hi%(loo
p%)=(11-loop%)*100:hi$(loop
%)="Electron":NEXT
205ENVELOPE1,129,-15,-8,-
3,10,10,10,126,0,0,-126,126
,126
206ENVELOPE2,1,6,6,6,2,2,
1,126,0,0,-126,126,126
207ENVELOPE3,1,1,-2,1,5,5
,12,126,0,0,-126,126,126
208RESTORE210
209FORloop%=0TO3:READchr1
%,chr2%,chr3%:man$(loop%,1)
=CHR$(chr1%)+CHR$10+CHR$8+C
HR$(chr2%):man$(loop%,2)=CH
R$(chr1%)+CHR$10+CHR$8+CHR$
(chr3%):NEXT
210DATA255,254,253,252,25
1,250,249,248,248,249,248,2
48
211bl$=" "+CHR$10+CHR$8+"
 "
212ENDPROC
213REM Asseable Machine c
ode
214DEF PROCassemble
215xp%=&70:yp%=&80:di%=&9
0:dead%=&A0
216scr%=&80:rnd%=&81:numb
er%=&82
217xc%=&83:mhit%=&84
218OS=&FFEE:?number%=7
219DIM code_space 300
220FOR PASS=0 TO 2 STEP 2
221P%=code_space
222[ OPT PASS
223.CODE LDXnumber%:LDA#1
7:JSROS:LDA#2:JSROS
224.cont LDAdead%,X:CMP#1
:BEQdecrem:LDA#31:JSROS:LDA
xp%,X:JSROS:LDAyp%,X:JSROS:
LDA#32:JSROS
225LDAdi%,X:CMP#1:BEQleft
:CMP#2:BEQright:CMP#3:BEQdo
wn:CMP#4:BEQup
226.print LDA#31:JSROS:LD
Axp%,X:JSROS:LDAyp%,X:JSROS
:STXxc%:LDA#135:JSR&FFF4:ST
Xscr%:LDXxc%:LDAscr%:CMP#32
:BNEhit
227LDA#246:JSROS:.dec DEX
:TXA:CMP#0:BEQfinish
228JMPcont
229.decrem JMPdec
230.left DECxp%,X:JMPprin
t
231.right INCxp%,X:JMPpri
nt
232.down INCyp%,X:JMPprin
t
233.up DECyp%,X:JMPprint
234.finish JMPover
235.hit LDAscr%:CMP#152:B
CSmehit
236.ht2 LDYdi%,X:LDArnd%:
STAdi%,X:TYA:CMP#1:BEQright
:CMP#2:BEQleft:CMP#3:BEQup:
CMP#4:BEQdown
237.mehit LDA#1:STAmhit%:
JMPht2
238.over LDA#17:JSROS:LDA
#3:JSROS:RTS
239]
240NEXT PASS
241ENDPROC
242REM Place the aliens
243DEF PROCaliens:COLOUR2
244FORloop%=1 TO ?number%
245xa%=RND(16)+2:ya%=8+RN
D(18)
246IFFNscrn(xa%,ya%)<>32
THEN 245
247VDU31,xa%,ya%,246:?(xp
%+loop%)=xa%
248?(yp%+loop%)=ya%:?(dea
d%+loop%)=0
249?(di%+loop%)=RND(4):NE
XT
250?mhit%=0:ENDPROC
251REM Place objects on s
creen
252DEF PROCobjects(amount
%,character%,colour%)
253IFcolour%=4THENchange%
=TRUE ELSEchange%=FALSE
254COLOURcolour%:FORloop%
=1TOamount%
255xpos%=RND(18):ypos%=RN
D(17)+9
256IFxpos%=17ANDypos%<9TH
EN255
257IFFNscrn(xpos%,ypos%)<
>32THEN255
258IFchange% VDU17,RND(3)
259VDU31,xpos%,ypos%,char
acter%
260NEXTloop%:ENDPROC
261IFERR=17THEN26
262REM ERROR!!!!!
263VDU7
264MODE6
265*FX4,0
266REPORT:PRINT" at line
";ERL
267*FX12,0
268*FX15,0
269END
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

Maths Test listing

From Page 31

```
630 PRINT "  ANSWER  ";
    :D=FN_get_number(SP%
    ,TRUE )
640 PROCans
650 PROCdelay(1)
660 UNTIL TIME >=ENDTIME

670 ENDPROC
680 REM **********************
    *******
690 REM SUBTRACTION
700 REM **********************
    *******
710 DEF PROCsub
720 *FX15,1
730 CLS
    :COLOUR 3
    :PRINT TAB(7,15)"READY?"
    :PROCdelay(2)
740 T%=TIME
    :ENDTIME=T%+MF
750 REPEAT
760 CLS
770 REPEAT
780 A=RND(NUMBER)
    :B=RND(NUMBER)
790 UNTIL A>B
800 C=A-B
810 COLOUR 2
    :PRINT TAB(2,10)"WHAT
     IS ";A;" - ";B
820 PRINT '''''
830 PRINT "  ANSWER  ";
    :D=FN_get_number(SP%
    ,TRUE )
840 PROCans
850 PROCdelay(1)
860 UNTIL TIME >= ENDTIME
870 ENDPROC
880 REM **********************
    *******
890 REM MULTIPLICATION
900 REM **********************
    *******
910 DEF PROCtimes
920 *FX15,1
930 CLS
940 REPEAT
950 COLOUR 3
    :PRINT TAB(2,10)"WHICH
     TABLES"
960 PRINT TAB(2,12)"BETWEEN
     2 AND 20?"
970 PRINT TAB(2,18)"YOUR
    CHOICE?";
980 B=FN_get_number(1000
    ,FALSE )
990 IF B<2 OR B>20
    THEN PRINT TAB(2,18)
    "NICE ONE TRY 2-20"
    :PROCdelay(1)
    :PRINT TAB(2,18)STRING$(2
    0," ")
1000 UNTIL B>2 AND B<21
1010 T%=TIME
    :ENDTIME=T%+MF
1020 CLS
    :COLOUR 3
    :PRINT TAB(7,15)"READY?"
    :PROCdelay(2)
1030 *FX15,1
1040 REPEAT
1050 A=RND(NUMBER)
1060 C=A*B
1070 CLS
1080 COLOUR 128
    :COLOUR 1
    :CLS
1090 PRINT TAB(2,10)"WHAT
     IS ";A;" X ";B
1100 PRINT '''''
1110 PRINT "  ANSWER  ";
    :D=FN_get_number(SP%
    ,TRUE )
1120 PROCans
1130 PROCdelay(1)
1140 UNTIL TIME >= ENDTIME
1150 ENDPROC
1160 REM **********************
    *******
1170 REM ANSWER ROUTINE
1180 REM **********************
    *******
1190 DEF PROCans
1200 CLS
1210 IF time%=TRUE
    THEN COLOUR 3
    :PRINT TAB(2,16)"TOO
     SLOW MUPPET!"
    :PROCsound_no_ans
    :W=W+1
    :ENDPROC
1220 IF D=C
    :COLOUR 11
    : PRINT TAB(7,16)"CORRECT
    "
    :PROCsound_right
    :OSBYTE=&FFF4
    :A%=9
    :X%=4
    :CALL OSBYTE
    :A%=10
    :CALL OSBYTE
    :R=R+1
    :SCORE=SCORE+(10-SPEED)
    :ENDPROC
1230 IF D<>C
    THEN COLOUR 6
    :PRINT TAB(7,16)"WRONG"
    :PRINT TAB(2,18)"THE
     ANSWER IS ";C
    :PROCsound_wrong
    :W=W+1
1240 *FX9,25
1250 *FX10,25
1260 ENDPROC
1270 REM **********************
    *******
1280 REM **********************
    *******
1290 DEF PROCsound_right
1300 SOUND 1,-15,200,4
1310 ENDPROC
1320 REM **********************
    *******
1330 REM **********************
    *******
1340 DEF PROCsound_wrong
1350 SOUND 1,-15,2,4
1360 ENDPROC
1370 REM **********************
    *******
1380 REM **********************
    *******
1390 DEF PROCsound_no_ans
1400 SOUND 1,-15,2,4
    :SOUND 2,-15,3,4
    :SOUND 0,-15,2,8
1410 ENDPROC
1420 REM **********************
    *********
1430 REM INSTRUCTIONS
1440 REM **********************
    *********
1450 DEF PROCinst
1460 COLOUR 130
    :CLS
1470 COLOUR 0
    :PRINT TAB(9,1)"##########
    ##########"
1480 PRINT TAB(9,2)"#
                    #"
1490 PRINT TAB(9,3)"#
                    #"
1500 PRINT TAB(9,4)"#
                    #"
1510 PRINT TAB(9,5)"##########
    ##########"
1520 VDU 5
1530 VDU 19,3,11,0,0,0
    :GCOL 0,3
1540 MOVE 340,926
    :PRINT "M A T H S   T
     E S T"
1550 VDU 4
1560 PRINT TAB(6,7);"THIS
     GAME IS DESIGNED TO
     TEST"
1570 PRINT TAB(6,9);"YOUR
     SPEED AND ABILITY TO
     ADD"
1580 PRINT TAB(6,11);"SUBTRACT
     AND MULTIPLY NUMBERS."
1590 COLOUR 1
    :PRINT TAB(6,14)"YOU
     MAY....."
1600 PRINT TAB(6,16);"1.CHOOSE
     YOUR SPEED. ( 1 - 5
     )"
1610 PRINT TAB(6,18);"2.CHOOSE
     YOUR RANGE. ( 2 - 99
     )"
1620 PRINT TAB(6,20);"3.CHOOSE
     YOUR OPERAND.(+ - *)"
1630 COLOUR 0
    :PRINT TAB(6,23);"YOU
     HAVE ONE MINUTE TO ANSWE
     R"
1640 PRINT TAB(6,25);"AS MANY
     QUESTIONS AS POSSIBLE."
1650 PRINT TAB(6,27);"MORE
     SPEED.....MORE MARKS
     "
1660 COLOUR 11
    :PRINT TAB(8,30);"PRESS
     ANY KEY TO CONTINUE"
1670 A=GET
1680 CLS
1690 COLOUR (1)
    :PRINT TAB(4,4)"YOUR
     SPEED"
1700 PRINT TAB(4,5)".....
    ."
1710 COLOUR 4
    :PRINT TAB(6,7)"The numbe
    r you enter will set"
1720 PRINT TAB(6,9)"the length
     of time that"
1730 PRINT TAB(6,11)"each
     sum stays on the screen.
     "
1740 PRINT TAB(6,13)"E.G."

1750 PRINT TAB(6,15)"5 gives
```

→

Maths Test listing

From Page 57

```
     10secs  for 1 point"
1760 PRINT TAB(6,17)"1 gives
      2sec   for 5 points"
1770 PRINT TAB(6,19)"The faste
     r you play....."
1780 PRINT TAB(6,21)"the more
     points you can score"
1790 COLOUR 0
     :PRINT TAB(6,24)"IF YOU
      DO NOT ANSWER IN TIME"
1800 PRINT TAB(6,26)"YOU SCORE
      AS A WRONG ANSWER."
1810 COLOUR 11
     :PRINT TAB(7,29)"PRESS
      ANY KEY TO CONTINUE"
1820 A=GET
1830 CLS
1840 COLOUR 1
     :PRINT TAB(4,6)"YOUR
      RANGE"
1850 PRINT TAB(4,7)".....
     "
1860 COLOUR 4
     :PRINT TAB(6,9)"The numbe
     r you enter will set"
1870 PRINT TAB(6,11)"the highe
     st WHOLE number"
1880 PRINT TAB(6,13)"you wish
      to use in your maths."
1890 PRINT TAB(6,15)"You score
      a bigger bonus"
1900 PRINT TAB(6,17)"for a
      larger range of numbers.
     "
1910 PRINT TAB(6,19)"E.G."
1920 PRINT TAB(6,21)"2 - 10
            scores 10"
1930 PRINT TAB(6,23)"2 - 20
            scores 20"
1940 PRINT TAB(6,25)"and so
      on....."
1950 COLOUR 11
     :PRINT TAB(8,30)"PRESS
      ANY KEY TO START"
1960 ENDPROC
1970 REM ********************
     ****
1980 REM SPEED INPUT
1990 REM ********************
     ****
2000 DEF PROCspeed
2010 *FX15,1
2020 CLS
     :COLOUR 131
2030 PRINT TAB(2,10)"SELECT
      YOUR SPEED"
2040 PRINT TAB(2,12)"5 SLOW
           1 FAST"
2050 REPEAT
     :COLOUR 4
     :PRINT TAB(2,18)"YOUR
      CHOICE?   ";
     :SPEED=FN_get_number(1000
     0,FALSE )
2060 IF SPEED >5 OR SPEED<1
     PRINT TAB(1,18)"GOOD
      START TRY 1-5!"
     :PRINT TAB(0,19)STRING$(2
     0," ")
     :PROCdelay(1)
     :PRINT TAB(0,18)STRING$(2
     0," ")
2070 UNTIL SPEED<6 AND SPEED>0
2080 PRINT TAB(2,24);"YOU
      GET "(6-SPEED)"  FOR"

2090 PRINT TAB(2,26)"A CORRECT
      ANSWER"
2100 SP%=SPEED*SF
2110 PROCdelay(2)
2120 ENDPROC
2130 REM ********************
     *****
2140 REM RANGE INPUT
2150 REM ********************
     *****
2160 DEF PROCnumbers
2170 *FX15,1
2180 CLS
2190 COLOUR 0
     :PRINT TAB(2,8)"TO SET
      THE TOP"
2200 PRINT TAB(2,10)"OF YOUR
      RANGE,"
2210 PRINT TAB(2,12)"SELECT
      A NUMBER"
2220 PRINT TAB(2,14)"BETWEEN
      2 AND 99"
2230 COLOUR 4
     :REPEAT
     :PRINT TAB(2,18)"YOUR
      CHOICE?  ";
     :NUMBER=FN_get_number(100
     00,FALSE )
2240 IF NUMBER>99 OR NUMBER<2
      PRINT TAB(2,18)"NEARLY,
       TRY 2-99!"
     :PRINT TAB(0,19)STRING$(3
     2," ")
     :PROCdelay(1)
     :PRINT TAB(2,18)STRING$(2
     0," ")
2250 UNTIL NUMBER>1 AND NUMBER
     <=99
2260 BONUS=INT (NUMBER/10+1)*1
     0
2270 PRINT TAB(2,24)"BONUS
      WILL BE ";BONUS
2280 PROCdelay(2)
2290 ENDPROC
2300 REM ********************
     *****
2310 REM MENU
2320 REM ********************
     *****
2330 DEF PROCmenu
2340 COLOUR 128
     :CLS
2350 COLOUR 14
     :PRINT TAB(0,6)"SELECT
      THE OPERATION"
2360 COLOUR 4
     :PRINT TAB(4,13);"1.
        ADD"
2370 COLOUR 2
     :PRINT TAB(4,16);"2.
        SUBTRACT"
2380 COLOUR 3
     :PRINT TAB(4,19);"3.
        MULTIPLY"
2390 ENDPROC
2400 REM ********************
     **********
2410 REM SCORE SHEET
2420 REM ********************
     **********
2430 DEF PROCscore
2440 *FX15,1
2450 COLOUR 131
     :CLS
2460 COLOUR 0
2470 PRINT TAB(0,2)"YOU GOT
      ";R;" RIGHT"
2480 PRINT TAB(4,4)"AND  ";W;
     " WRONG"
2490 PRINT TAB(0,8)"YOUR BONUS
      WAS ";BONUS
2500 SCORE=SCORE+BONUS
2510 COLOUR 4
     :PRINT TAB(0,14)"TOTAL
      SCORE ";SCORE
2520 PRINT TAB(0,15)"=========
     ======"
2530 IF SCORE>HI
     THEN HI=SCORE
2540 COLOUR 1
     :PRINT TAB(0,20)"HIGH
      SCORE ";HI
2550 COLOUR 11
     :PRINT TAB(0,26)"TRY
      AGAIN ?"
2560 A$=GET$
2570 IF A$="Y"
     THEN ENDPROC
2580 VDU 20
2590 END
2600 ENDPROC
2610 REM ********************
     **********
2620 REM NUMBER INPUT ROUTINE
2630 REM ********************
     **********
2640 DEF FN_get_number(DL%
     ,TL%)
2650 final$=""
2660 TIMENOW=TIME
2670 TIME =0
2680 REPEAT
2690 REPEAT
2700 temp=INKEY (0)
2710 time%=(TIME >DL%
     AND TL%)
2720 UNTIL (temp>=48 AND temp<
     =57) OR temp=13 OR temp=1
     27 OR time%
2730 IF temp<>13 AND temp<>127

     THEN final$=final$+
     CHR$ (temp)
2740 IF LEN (final$)<>0
     THEN VDU temp
2750 IF temp=127 AND LEN (fina
     l$)>0
     THEN final$=LEFT$(final$
     ,LEN (final$)-1)
2760 UNTIL ((temp=13 OR temp=-
     1)AND final$<>"")
     OR time%
2770 IF temp=-1 final$="-999"
2780 TIME =TIMENOW+TIME
2790 =EVAL (final$)
2800 REM ********************
     **********
2810 REM DELAY
2820 REM ********************
     **********
2830 DEF PROCdelay(SECS%)
2840 LOCAL time
2850 time=TIME
2860 REPEAT UNTIL TIME >=time+
     DF*SECS%
2870 ENDPROC
```

This listing is included in this month's cassette tape offer. See order form on Page 47.

ADVERTISERS INDEX

Amplesoft	15
British Telecom	2
DACC	60
Doctor Soft	16
Epic Software	52
First Byte	4
Golem	30
Haystack Peripherals	52
Kay-Ess	20
Kosmos Software	30
M.P. Software	60
Micropower	64
Mushroom Computers	13
National Micro Centres	22,23
Newnes Tech Books	30
Pace	49
Qualsoft	59
S.C.I. UK	55
Signpoint	11
Simonsoft	44
Sir Computers	20
Slogger Software	59
Squirrelsoft	52
Superior Software	63
T.S.L. Ltd	60

Micro Messages

Spooling can save that precious program

A USEFUL procedure when typing in long programs (provided that you have motor control) is to save them while typing them in as an Ascii text file.

*You do this by placing a blank tape in the recorder, type in *SPOOL "name" and then press Return.*

The message "Press record then Return" will appear on screen. After obeying this just type in the program as normal.

After each few lines the recorder will start and nothing more will be written on the screen until it stops.

*After a typing session type in *SPOOL and press Return to close the file and leave the tape in the same position. On the next programming session repeat the above procedure.*

*To retrieve the completed program rewind the tape to the beginning of the spooling. Then *EXEC"" each section provided that all the line numbers are consecutive.*

When this has been done the complete program can be listed and saved in the normal way.

Spooling like this takes longer than the normal saving and loading but it can prevent a program being lost due to power failures and other disasters. – **K. Goodacre, Sheffield.**

● Many thanks for the tip. Is there any one of us who hasn't lost hours of work that might have been saved by this method? Has anyone else any useful tips and hints?

Move into disc space

I BOUGHT a Plus 1 last week and immediately noticed something rather odd. It uses a bit of the disc space (or, rather, the disc space used by the BBC OS) poking a few numbers in at &D68 to &D6F and &DBA to &DBC.

If you fill this with any data or if you assemble a machine code routine here, you cannot load or save a program. Ctrl/Break restores normal operation. – **Roland Waddilove, Widnes, Cheshire.**

● Obviously Acorn are using that page of memory for something, probably the joysticks. We wonder what they'll do when and if they get round to using discs.

Priority repair?

IN the April issue of Electron User reference was made to Acorn's high failure rate. I am sad to say my machine was one of the eight to 25 per cent which entered the failure mode.

However I wonder what Acorn consider a priority repair, as I had to make several phone calls and wait six weeks before they eventually managed to return my machine.

Anybody else had this problem? – **L.D. Wright, Livingston, West Lothian.**

● You seem to have struck unlucky, Mr Wright. Certainly, apart from one other complaint, all we've heard is praise for the way that Acorn have dealt with faulty Electrons.

```
10 REM LE CHAT
20 REM by David Kennelly

30 MODE 2
40 VDU 23,224,36,24,27
   ,254,190,36,102,0
50 VDU 23,225,36,24,192
   ,318,189,36,102,0
60 VDU 23,1,0,0;0;0;
70 VDU 9
   :COLOUR 3
   :VDU 224
80 VDU 8
   :VDU 8
   :COLOUR 0
   :VDU 225
90 SOUND 1,-15,197,1
100 FOR A=0 TO 500
   :NEXT
110 VDU 9
   :COLOUR 3
   :VDU 225
120 VDU 8
   :VDU 8
   :COLOUR 0
   :VDU 224
130 SOUND 1,-15,205,1
140 FOR H=0 TO 500
150 NEXT
160 GOTO 70
```

Dancing cat

I HAVE called this program Le Chat. It uses one of your Casting Agency characters and a variant of it.

The program makes the cat dance across the screen, slowly moving downwards. – **David Kennelly.**

Bus poser

PERHAPS J. Williamson's case (Micro Messages, June 1984) isn't so untypical. My Electron intermittently failed to respond to the line of keys 9, 0, I and fullstop.

This seemed to be due to something wrong with the bus connector from the keyboard to motherboard. A slight "play" with this connector cured the problem.

A pity that a connector problem should let down such a good product. – **Mike Arnold, Worcester.**

● We know of another Electron which had exactly the same problem and wonder how many other owners have suffered too. Incidentally, opening the case and messing around will probably invalidate the six month warranty.

WHAT would you like to see in future issues of Electron User?

What tips have you picked up that could help other readers?

Now's here is your opportunity to share your experiences.

Remember that these are the pages that you write yourselves. So tear yourself away from your Electron keyboard and drop us a line.

The address is:

Micro Messages
Electron User
Europa House
68 Chester Road
Hazel Grove
Stockport
SK7 5NY.

Memory options

MY congratulations on starting a really superb magazine. Its helped me understand my Electron with ease.

Could you tell me if the Acorn Electron has any other

→

Micro Messages

From Page 61

memory options available? Also is it true that Acorn are bringing out a Mode 7 add-on for the Electron? – **Manjit Singh Ahir (age 13), Willenhall, West Midlands.**

● At present, there are no other memory options available. Also, despite some initial speculation, no one has, as yet, figured out a way of providing Mode 7 for the Electron.

Any advance?

COULD you tell me if there is an Electron version of the Advanced User Guide? My friends who have BBC Micros swear by it.

If there isn't an Electron version, does the BBC book apply to the Electron? – **Alan Sargeant, Stockport.**

● There isn't an Advanced User Guide for the Electron. We asked one of the programmers who developed the Electron whether or not the BBC version would apply to it. His answer was that it would, "apart from the bits that don't".

For those who find the Advanced Guide hard going we recommend the excellent "Electron and BBC Basic, Quick Reference Guide for Programmers".

Written by Alison Carling and published by DP Publications, it's well worth having a copy.

```
 5 MODE 0
 6 LET C=FALSE
 7 VDU 23;8202;0;0;0;
10 VDU 29,200;300;
11 MOVE 0,0
20 FOR X=0 TO 6.3 STEP 0.01
30 LET P=COS (10*X)*
   SIN (5*X)
40 LET Y=COS (10*X)*
   COS (X)
50 P=P*130+159
60 Y=Y*90+95
65 LET P=P*3
   :LET Y=Y*3
70 DRAW P,Y
75 IF C=FALSE
   THEN CLS
   :C=TRUE
80 NEXT
```

Graphic puzzle

I THOUGHT that you might be interested in the above graphics program.

C.Hill (no address)

● Many thanks for the program, Mr Hill. It is nice, but it leaves us with one question. What is it?

Palette problem

I'VE had my Electron some months now but I cannot find out how to get the colour brown. I know that you can get this colour because the ape in Program Power's Killer Gorilla is brown.

So far all my attempts have failed, so could you please tell me how. – **A. Fox, Buxton, Derbyshire.**

● We suspect that you could get brown by using the colours red, yellow and blue in the same user defined characters. Has anyone got a better method using Basic or even machine code?

Powerful command

WHEN I finally dragged myself away from a thoroughly enjoyable game I decided to do something constructive... Eventually I came up with a short program called Handy Calc (below) which makes use of the EVAL command.

I think that it might encourage other readers to use this immensely powerful command in their programs. – **Gary Fitton, Milton Keynes.**

● Many thanks, Gary. As you so rightly say, EVAL is a very powerful feature of Electron Basic and one that isn't used as much as it could be.

Looking for a disc

I RECENTLY bought an Acorn Electron and I would like to know if all Electrons do what mine does when you press Shift and Break at the same time. Mine prints the message "Searching" in between Acorn Electron and Basic.

On the next line is the message "File not found". It doesn't worry me – I'd just like to know why it prints it. – **M. Edes, Folkestone, Kent.**

● You do right not to worry. As you know, the Electron operating system is a copy of the BBC's.

On the BBC machine when you hold down the Shift key, then press and release the Break key (still holding down the Shift) the micro immediately goes searching for a file on a disc. Since the Electron doesn't have a disc you get the messages.

```
10 REM Handy Calc
20 REM by Gary Fitton
30 ON ERROR GOTO 110
40 MODE 6
50 PRINT TAB(0,3)"Handy
   Calc by G.Fitton"
60 PRINT TAB(0,4)"----------
   ----------"
70 INPUT ''"QUESTION",A$
80 A=EVAL (A$)
90 PRINT '' "ANSWER= ";A
100 PRINT TAB(0,20)"PRESS
    ANY KEY FOR ANOTHER
    GO"
    :Z=GET
    :RUN
110 PRINT TAB(0,15)"SORRY,
    CAN'T HELP YOU"TAB(0
    ,20)"PRESS ANY KEY FOR
    ANOTHER GO"
    :Z=GET
    :RUN
```

SPECIAL OFFER!
Deduct £1 per cassette when ordering 2 or more.

TOP QUALITY SOFTWARE FOR THE ACORN ELECTRON

ACORN ELECTRON

PERCY PENGUIN **£7.95**
The best version available for the Electron micro. Percy is trapped in an ice maze which is populated by the deadly Snobees. His only hope of survival is to squash them by hurling ice cubes at them. Unfortunately, whenever it seems that he has won, a deadlier breed appears. Hi-score, rankings, excellent graphics and sound.
NEW RELEASE

MR. WIZ **£7.95**
From the author of Percy Penguin, Mr. Wiz is a fast-action multi-scene game. Guide Mr. Wiz around the garden to eat the cherries whilst avoiding the evil gremlins. The gremlins can be killed by dropping apples on them or by throwing the crystal ball. Extra points can be gained by eating the magic mushroom, but beware...this is the home of the gremlins and makes them permanently furious! Sound effects and tunes, hi-score, rankings. Superb arcade-style action.
NEW RELEASE

CHESS **£7.95**
A highly versatile implementation of Chess. Play black or white against the computer or a human opponent. The skill level of the computer's play can be varied widely, and moves are entered either by co-ordinates, cursor control, or joystick control. Moves can be taken back if an error has been made, and the board can be modified at any time. Games can be "saved" or "loaded", and the last game can be replayed. The computer will, if requested, suggest your moves.
NEW RELEASE

CENTIBUG **£7.95**
The centibug descends from the top of the screen weaving intimidatingly between the mushrooms. Your objective is to shoot all the segments of the centibug before it reaches the bottom of the screen.
Features include: spiders, snails, flies, 6 skill levels, hi-score, rankings, and increasing difficulty.

ALIEN DROPOUT **£7.95**
A novel and unusual program. Arcade-action with this exciting multi-stage shooting game. The objective of the game is to shoot the aliens out of their "boxes" before the "boxes" fill up. Once full, the aliens fly down relentlessly, exploding as they hit the ground. The game features include: 6 skill levels, rankings, hi-score, increasing difficulty.

STRANDED **£7.95**
An adventure game using hi-resolution full-colour graphics. You are stranded on a strange planet, and your mission is to return to civilisation and home. Many of the locations are shown graphically, including the spaceship, the cliffs, the mountains, and (if you succeed) your home. You must carefully explore your environment searching for hidden clues to help you in your quest.
NEW RELEASE

WORLD GEOGRAPHY **£7.95**
This program covers 166 countries which are divided into 8 categories of difficulty. Each country is pinpointed on an accurate hi-resolution screen map of the world, and the user is asked the capital and/or population. At the end of the test, the percentage of correct answers is given, so that the student can monitor his geographical knowledge.

ALSO AVAILABLE:

INVADERS	£7.95	DISASSEMBLER	£7.95
FRUIT MACHINE	£7.95	DRAUGHTS	£6.95
CONSTELLATION	£7.95	REVERSI	£6.95

DEALERS – Our software is now available at all good dealers including: selected branches of W. H. Smith and Boots; all major computer dealers – Microstyle, Electronequip, 3D Computers, Computerama, GTM Computers, etc.; and our software is also available through all the major distributors, and directly from us.

WE PAY UP TO 20% ROYALTIES FOR HIGH QUALITY BBC MICRO AND ELECTRON PROGRAMS.

SUPERIOR SOFTWARE LTD.
Dept. EU8, Regent House,
Skinner Lane, Leeds 7
Tel: 0532 459453

OUR GUARANTEE

(1) All our software is available before we advertise.
(2) All our software is despatched within 48 hours by first-class post.
(3) In the unlikely event that any of our software fails to load, return your cassette to us and we will immediately send a replacement.


MICROPOWERMICROPOWERM
GHOULS
commodore 64
(SUPER-FAST LOADING TIME)
electron
B.B.C. MICRO
Run through the
creepy mansion
to rescue the
power jewels. Dodge ghostly
ghouls and bouncing spiders, leap
over poison-smeared spikes,
scamper along moving platforms
and contracting floorboards,
and use erful springs to
propel to overhanging
dges b animation
d spi gling
und effe
.95.
(ELECTRON and
BBC MICRO
versions
£7.95)
COMMODORE 64 VERSION
MICRO
POWER
TAKE MY
ADVICE! BOOST YOUR
COLLECTION WITH SWOOP,
FELIX IN THE FACTORY
AND CYBERTRON
MISSION!
MICRO POWER LTD.,
NORTHWOOD HOUSE, NORTH STREET,
LEEDS LS7 2AA TEL: (0532) 458800
SELECTIVE BRANCHES OF BOOTS, CO-OP,
MENZIES, W. H. SMITH, WOOLWORTHS AND ALL
GOOD DEALERS.
AUTHORS! WE PAY 20% ROYALTIES!